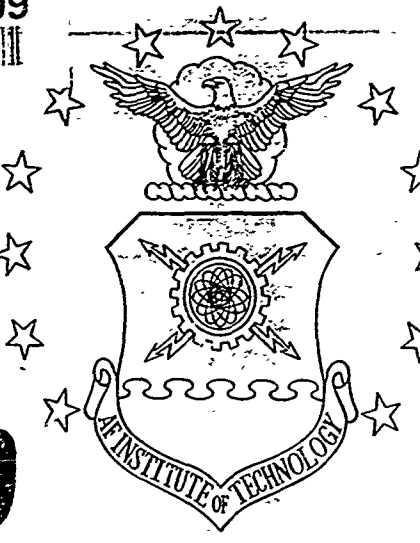


DEPARTMENT OF THE AIR FORCE

AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GCE/ENG/91D-02

THE APPLICATION OF STATISTICAL ESTIMATION TECHNIQUES TO TERRAIN MODELING

THESIS

Donald P. Duckett, Jr.
Captain, USAF

AFIT/GCE/ENG/91D-02

Approved for public release; distribution unlimited

AFIT/GCE/ENG/91D-02

THE APPLICATION OF STATISTICAL ESTIMATION TECHNIQUES TO TERRAIN MODELING

THESIS

Presented to the Faculty of the School of Engineering

of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Computer Engineering

Donald P. Duckett, Jr., BS

Captain, USAF

December, 1991

Approved for public release; distribution unlimited

Acknowledgments

This thesis is the result of the efforts of many fine people. My appreciation goes out to fellow students Capt Chris Brodkin and Capt Wayne McGee for their help in learning and implementing the kriging methods, and to Maj David Robinson for fulfilling the role as the local kriging guru. I would also like to thank the other students in the graphics sequence here at AFIT for their continuous assistance and support, and my thesis committee (Lt Col Phil Amburn, Lt Col Marty Stytz, and Maj Paul Bailor) for their useful direction, insight and comments.

Thanks also goes out to my family in North Carolina and to all of my good Christian friends both at AFIT and at the First Baptist Church of Fairborn, Ohio. The prayers and support they provided over the eighteen months that I was in residence at AFIT were indispensable. However, my greatest thanks is to my Lord Jesus Christ, through whom all things are possible.

Donald P. Duckett, Jr.

Table of Contents

		Page
Acknowledgments		ii
Table of Contents		iii
List of Figures		vii
List of Tables		ix
Abstract		x
I.	Introduction	1-1
	1.1 Motivation	1-1
	1.2 Problem Statement	1-3
	1.3 Approach	1-3
	1.4 Definitions	1-4
	1.5 Research Objectives and Scope	1-5
	1.6 Limitations	1-6
	1.7 Organization	1-7
II.	Background on Terrain Modeling and Kriging	2-1
	2.1 Terrain Modeling For Flight Simulators	2-1
	2.1.1 Terrain Model Sources.	2-1
	2.1.2 Terrain Modeling Techniques.	2-2
	2.1.3 General Terrain Modeling.	2-6
	2.1.4 Summary	2-6
	2.2 Kriging	2-7
	2.2.1 Origin of Kriging.	2-7

Page

2.2.2 Definition of Kriging. 2-8

2.2.3 Kriging Equations. 2-10

2.2.4 Neighborhoods 2-13

2.2.5 Kriging Assumptions. 2-14

2.2.6 Kriging with a Trend. 2-15

2.2.7 Structural Analysis. 2-17

2.2.8 Other Forms of Kriging. 2-23

2.2.9 Summary 2-24

2.3 Summary 2-25

III. Development of the Kriging Methodology for Terrain Modeling 3-1

3.1 Applicability of Kriging to Terrain Elevation Estimation . . . 3-1

3.2 Variogram Modeling 3-3

3.3 Kriging 3-5

3.4 Error Variance Minimization 3-7

3.5 Anisotropy 3-10

3.6 Summary 3-12

IV. Implementation of the Terrain Modeling Systems 4-1

4.1 System Requirements 4-1

4.2 System Overview 4-2

4.3 Phase I: Terrain Modeling using Filtering 4-3

4.3.1 Overview. 4-3

4.3.2 `tape2dted`. 4-3

4.3.3 `dted2pts`. 4-5

4.3.4 `pts2geom`. 4-6

4.3.5 `connect`. 4-9

4.3.6 `tb`. 4-10

Page

4.4 Phase II: Terrain Modeling using Kriging 4-10
4.4.1 Overview. 4-10
4.4.2 `resid`. 4-12
4.4.3 `varfit`. 4-13
4.4.4 `krige`. 4-14
4.4.5 `rebuild`. 4-16
4.4.6 `partition`. 4-17
4.5 Summary 4-18

V. Results 5-1
5.1 Terrain Modeling using Filtering Techniques 5-2
5.2 Terrain Modeling using Kriging 5-3
5.2.1 Kriging From Full Resolution DTED. 5-4
5.2.2 Kriging From Filtered DTED. 5-5
5.2.3 Kriging from a Partial DTED Cell. 5-11
5.3 Terrain Modeling using Minimization With Respect To Error Variance. 5-13
5.4 Summary 5-14

VI. Conclusions and Recommendations 6-1
6.1 Conclusions 6-1
6.2 Recommendations 6-3
6.3 Summary 6-5

Appendix A. Development of Kriging Equations A-1

Appendix B. Terrain Elevation Distributions B-1
B.1 Distributions of Raw Terrain Elevation Data B-1
B.2 Distributions of Residual Terrain Elevation Data B-23

Page

Appendix C. Terrain Elevation Variograms C-1

C.1 Variograms of the DTED Evaluation Cells C-1

C.2 Variograms of the Residuals from the DTED Evaluation Cells C-23

C.3 Variograms for DTED Test Cells C-45

C.4 Variograms for the Partial DTED Cells Used in Chapter V . C-50

Appendix D. .pts File Format D-1

D.1 .pts File Data D-1

D.2 .pts File Header D-2

Appendix E. User's Manuals E-1

Bibliography BIB-1

Vita VITA-1

List of Figures

Figure		Page
2.1.	Typical Experimental Variogram Generated From Terrain Data	2-19
2.2.	Sample Linear Model Variogram	2-20
2.3.	Sample De Wijsian Model Variogram	2-20
2.4.	Sample Spherical Model Variogram	2-21
3.1.	Variogram Regularization Angle ϕ and Semi-Inclusion Angle ψ	3-4
4.1.	Phase I Pipeline	4-4
4.2.	Phase II Pipeline	4-11
5.1.	The Effect of Trend in Kriging – Original Grid	5-8
5.2.	The Effect of Trend in Kriging – Kriged Grid at 21 × 21	5-9
5.3.	The Effect of Trend in Kriging – Kriged Grid at 20 × 20	5-9
5.4.	Terrain Model Built by Filtering DTED Test Cell #1	5-15
5.5.	Terrain Model Built by Filtering DTED Test Cell #2	5-15
5.6.	Terrain Model Built by Filtering DTED Test Cell #3	5-16
5.7.	Terrain Model Built by Filtering DTED Test Cell #4	5-16
5.8.	Terrain Model Built from a $10° \times 10°$ Portion of DTED Test Cell #1 (Full Resolution)	5-17
5.9.	Terrain Model Built from a $10° \times 10°$ Portion of DTED Test Cell #2 (Full Resolution)	5-17
5.10.	Terrain Model Built from a 10° × 10° Portion of DTED Test Cell #1 (Filtered)	5-18
5.11.	Terrain Model Built from a 10° × 10° Portion of DTED Test Cell #2 (Filtered)	5-18
5.12.	Image of Multiple-Level-of-Detail Terrain Model Built from DTED Test Cell #1	5-19

Figure Page

5.13. Terrain Model Built by Filtering then Kriging DTED Test Cell #1 . . . 5-20

5.14. Terrain Model Built by Filtering then Kriging DTED Test Cell #2 . . . 5-20

5.15. Terrain Model Built by Filtering then Kriging DTED Test Cell #3 . . . 5-21

5.16. Terrain Model Built by Filtering then Kriging DTED Test Cell #4 . . . 5-21

5.17. Elevation-Shaded 2D Image of Non-Residual (left) and Residual Data (right) from DTED Test Cell #1 5-22

5.18. Elevation-Shaded 2D Image of Non-Residual (left) and Residual Data (right) from DTED Test Cell #2 5-22

5.19. Elevation-Shaded 2D Image of Non-Residual (left) and Residual Data (right) from DTED Test Cell #3 5-23

5.20. Elevation-Shaded 2D Image of Non-Residual (left) and Residual Data (right) from DTED Test Cell #4 5-23

5.21. Terrain Model Built by Kriging a 10° × 10° Portion of DTED Test Cell #1 (Full Resolution) 5-24

5.22. Terrain Model Built by Kriging a 10° × 10° Portion of DTED Test Cell #2 (Full Resolution) 5-24

List of Tables

Table		Page
1.1.	DTED Test Cells (Latitude & Longitude Correspond to Southwest Corner of Cell)	1-6
2.1.	DMA DTED Post Intervals	2-2
3.1.	DTED 1° × 1° Evaluation Cells (Latitude & Longitude is of Southwest Corner)	3-2
5.1.	DTED Test Cells	5-1
5.2.	Error Variances for Kriged Terrain Models	5-10
5.3.	Error Variances for Kriged Terrain Models Based on Partial DTED Cells	5-12

Abstract

This thesis researches methods of generating accurate and realistic polygonal terrain models by reducing gridded sampled terrain elevation data such as DMA DTED. The terrain models generated should be applicable for use in flight simulators and other systems. Existing methods of terrain modeling are discussed, and limitations of these systems are presented.

The geostatistical estimation technique known as kriging is presented as a method of estimating terrain elevations at locations not provided in DMA DTED or other gridded terrain elevation elevation data. Kriging is an optimal interpolation method based on statistical analysis of the data. It also provides a measure of accuracy, the error variance, that allows some control over the accuracy of the resulting terrain model.

This estimation method is employed in a terrain modeling system that builds polygonal terrain models at any resolution. This system is described in this thesis, and the results of terrain modeled both by filtering the DMA DTED and by estimating elevations with kriging are presented. The technique as implemented is computationally very expensive and has therefore limited the results of this thesis effort. Also, terrain models that could be produced appeared smoother than their filtered counterparts. However, kriging still shows promise as a method of estimating terrain elevations for terrain models.

THE APPLICATION OF STATISTICAL ESTIMATION TECHNIQUES TO TERRAIN MODELING

I. Introduction

The primary purpose of this thesis is to develop and evaluate methods to reduce dense terrain elevation data in order to generate accurate and realistic polygonal terrain models for use in flight simulators and other systems. The methods developed are employed in a terrain modeling system that provides polygonal terrain models at any resolution, not only at resolutions obtained from sampling the original gridded data. These methods also provide a measure of accuracy of the resulting polygonal model as compared to the original elevation data. These modeling methods use a geostatistical technique known as kriging to estimate elevation values at locations that may not have been sampled. Kriging is an optimal interpolation method based on statistical analysis of the data. This first chapter briefly covers the motivation for this research and provides the problem statement and approach for this thesis. Definitions, research objectives, and the scope, limitations, and organization of the thesis are also presented.

1.1 Motivation

As the cost of pilot training increases, there is a continuing effort by many in the aerospace community to develop new and innovative ways to train pilots. Flight simulators provide this type of training. Although not designed to replace actual training time in the aircraft, flight simulators are a valuable training substitute that can enhance pilot training in a cost effective way (52)(49).

One approach to flight simulator development is to use expensive, special-purpose, high-performance (and usually proprietary) computer systems designed specifically for flight simulators; in fact, several commercial flight simulator developers use such an approach (15)(27:733-4)(45). Another approach is to use relatively inexpensive general purpose graphics workstations to drive the simulator display. The Air Force Institute of

Technology (AFIT) has taken this approach, developing a practical and inexpensive flight simulator using a Silicon Graphics workstation driving a display system known as a head-mounted display (HMD) (39)(16). There is also a current effort to develop an object-oriented flight simulator for the Silicon Graphics 4D series graphics workstations to drive a similar HMD (43)(35)(3).

The second approach discussed above, with its cost advantages, requires that certain compromises be made. One such compromise is between the realism and accuracy of the images rendered by the flight simulator and the image update rate. General purpose graphics workstations usually generate images based upon a polygonal model of the environment in which the simulator is flying, and the way in which this model is generated directly affects both the image realism and accuracy and the image update rate. The realism perceived in an image is usually measured by the amount of detail contained in the model used to generate the image. This measure is usually a function of the density of polygons in that model, as each singularly colored polygon provides a single "bit" of information to the image (31:13). Therefore, the more polygons used to generate a model, the more realistic an image rendered from that model is felt to be. On the other hand, the image update rate is constrained by a similar measure, that of the number of polygons rendered for each scene. A less complex the model usually yields a higher update rate. A direct but inverse relationship is therefore apparent between the realism and accuracy of an image and the image update rate. The model used by the flight simulator to generate images must contain enough detail to infer to the user that the environment is real, but if the model is too complex, the workstations may not be able to generate images and update the flight simulator display fast enough to relay the impression of unconstrained flight (52)(34)(20).

In most flight simulators, rendering the terrain model over which the pilot flies is one of the most time-intensive steps; therefore, many flight simulators reduce the complexity of the terrain model to improve performance (31)(52). Many terrain modeling systems base their terrain models upon the Defense Mapping Agency's (DMA's) Digital Terrain Elevation Data (DTED) (20)(50)(34)(40). A detailed polygonal terrain model built directly from a one degree by one degree cell of DMA DTED could require 2,880,000 triangular

polygons to model. Typical flight simulators may need to render polygons from many such cells for each image generated, and many such images may need to be generated each second to provide the illusion of smooth flight. Therefore, a terrain modeling technique is needed that can sufficiently reduce the complexity of sampled terrain elevation data, allowing a flight simulator to maintain a reasonable update rate, while preserving much of the realism and accuracy inherent in the original terrain database. Although progress has been made toward developing such a technique, models used in real-time applications often sacrifice realism for efficiency, and the most realistic models may take hours to compute a single scene (26:263).

Traditional methods of reducing the terrain model complexity usually rely on filtering or sampling the original gridded terrain elevation data to produce a coarser grid. There is no guarantee that, for any given DTED cell, these techniques preserve enough detail to produce a realistic image. Neither is there any measure of the accuracy of the terrain model obtained by using these techniques. Additionally, when a gridded terrain model is generated by sampling the original terrain elevation database, the finest adjustment to the model's resolution is in increments of the spacing between the original samples in that database. Therefore, the number of polygons in the resulting model may not provide the best performance for a given flight simulator.

1.2 Problem Statement

A method is needed to reduce the complexity of dense terrain elevation data such that accurate and realistic terrain models can be built.

1.3 Approach

This thesis explores the possibility of using the geostatistical estimation method known as kriging to reduce the complexity of dense terrain elevation data for the purpose of terrain modeling.

Kriging is an optimal interpolation or estimation method that provides a best, unbiased, linear estimate of the value of a regionalized variable at any position based solely

on sampled data from the vicinity of the point being estimated. "Best" in this context refers to the estimate with minimum error variance. Compared to other interpolation or convolution methods, kriging should provide a more accurate estimation of the elevation of the terrain surface at any unsampled point. This should allow accurate and realistic terrain models to be generated at any resolution, even at resolutions not directly corresponding with the spacing of the original grid. It should also provide an optimal method of estimating elevations over a regular grid from from non-gridded terrain elevation data. Additionally, the error variance can be used as a measure of accuracy of any resulting model. These mechanisms should provide a fine level of control over the accuracy, realism, and complexity of any terrain model being generated.

1.4 Definitions

The following terms are used extensively in this thesis. These definitions are provided to ensure their consistent use throughout the thesis.

Accurate and Realistic Terrain Models. The goal of this thesis is to produce accurate and realistic terrain models from terrain elevation data. Accuracy and realism are closely related terms. However, the term *accurate* will be used in conjunction with a measure of accuracy – a terrain model is only judged as accurate if there is a quantitative measure of its accuracy and if that measure indicates that the model is indeed accurate. On the other hand, *realism* is defined by this thesis as a subjective quality of a terrain model; it is measured by the "goodness" of the images produced from the models and how closely these images resemble the actual terrain as represented by the unmodified terrain elevation data (or the user's impression of the actual terrain). As discussed above, this quality measure is closely related to the amount of detail in the terrain model; *detail* relates to the features present in the original terrain elevation data (cliffs, peaks, gorges, etc.) that are retained by the model. Therefore, this thesis does not define detail as strictly based on the polygon density, but a higher polygon density usually yields more detail.

Resolution. A gridded terrain model is a model whose polygon vertices lie in a regular gridded pattern with respect to the x and y directions. *Resolution*, therefore, is the number

of such vertices in one direction with respect to the total length of the model in that direction. As discussed earlier, gridded terrain models are easily constructed from gridded elevation data; therefore, the resolution of the terrain model and the resolution of the gridded data from which it is extracted are terms that are used interchangeably in this thesis.

Complexity. A goal of this thesis is to reduce a volume of terrain elevation data such as DTED to create less complex but accurate and realistic terrain models. As the resolution of a polygonal terrain model is directly related to the resolution of the original elevation data, this data must be reduced in some fashion. *Reduction* refers to the process of characterizing a volume of data by a single value and replacing that volume with the new value. This process reduces the *complexity* of the original elevation data by reducing its volume and density. If a reduced value correctly characterizes the area that it is based on (as is desired with kriging), the complexity of the resulting terrain model can be reduced while maintaining the accuracy and realism of the original data.

1.5 Research Objectives and Scope

The research objectives of this thesis are as follows:

- To implement a terrain modeling system that generates a polygonal terrain model based upon filtering gridded DMA DTED data at regular intervals. The terrain models generated by any tool resulting from this research will be used by systems using Brunderman's Graphical Database Management System (GDMS)(3), including Brunderman's Database Generation System (DBGen) (3)(35), Simpson's object-oriented flight simulator (43), and Gerken's Battlefield Management System (18). As support for rendering models in multiple-levels-of-detail is desired in these systems (35), another objective of this thesis is to model the terrain in such a way to support this rendering method. Terrain models built from the four DMA DTED cells listed in Table 1.1 (hereafter referred to as the *DTED Test Cells*) are used to evaluate the performance of this terrain modeling system.

DTED test cell #1	36° N x 113° W (Grand Canyon)
DTED test cell #2	36° N x 118° W (Death Valley)
DTED test cell #3	38° N x 121° W (Sierra Nevada)
DTED test cell #4	37° N x 123° W (San Francisco Bay)

Table 1.1. DTED Test Cells (Latitude & Longitude Correspond to Southwest Corner of Cell)

- To show the applicability of kriging as a technique to estimate terrain elevations based on dense terrain elevation data. Kriging was first developed to estimate various characteristics of naturally occurring ore reserves (10), and the similarity between subterranean formations and the terrain surface itself are obvious; the terrain surface can be simply considered another geological surface. However, some analysis of the technique as it applies to terrain surfaces is needed to ensure there are no peculiarities in this application. Also, the specific procedures for the structural analysis and the kriging processes as they apply to terrain surfaces need to be developed.
- To show that the kriging procedures developed for reducing terrain elevation data produce accurate and realistic polygonal terrain models from DMA DTED and to determine the applicability of the error variance as a measure of accuracy of such models. Subjective measures such as the "goodness" of the images generated from the terrain model are used to determine kriging's acceptability. This technique is also evaluated using terrain models built from the four DTED Test Cells listed in Table 1.1.
- To implement these techniques in the terrain modeling system described in the first objective in order to create a robust terrain modeling system for flight simulators. Any resulting model is subjectively evaluated according to the quality of the model produced, the ease of its construction, and the effect and intuitiveness of its controlling parameters.

1.6 Limitations

The following is a list of limitations for this thesis:

- This thesis is not concerned with developing methods to render the terrain models in real time, as the terrain models built will be tested using the systems that use Brunderman's GDMS (3).
- Correspondingly, this thesis is not concerned with the accuracy desired for a specific system using the terrain model, nor is it concerned with the resolution necessary for that system's optimum performance. This thesis is simply concerned with developing tools to allows the creation of appropriate terrain models as determined by the requirements of the different systems.
- As model building for flight simulators is usually accomplished in a preprocessing step before the flight simulator is used, immediate turn-around is not required of the modeling system. However, excessively long turn-around times cannot be tolerated due to the limited development time for this thesis.
- The DMA DTED cells that are used in this thesis are defined to be 1 degree longitude by 1 degree latitude quadrangles with elevation posts every 3 arc seconds of longitude or latitude. Depending on the actual latitude, an arc second of longitude can vary tremendously in actual size, causing variations in the regularity of the elevation posts. However, this thesis assumes that, for the quadrangles used, all elevation posts are regularly spaced.
- Any subsystem in this terrain modeling system should handle data that may be either regularly gridded or not; however, the final resulting terrain model will be composed of regularly gridded polygons.
- Finally, as kriging is a stochastic estimation technique, this thesis does not attempt to prove that kriging is the best such technique for reducing dense terrain elevation data. However, some justification for its application to terrain modeling is provided in Chapter III.

1.7 Organization

This thesis is organized into five chapters and five appendices. Chapter II surveys terrain modeling and kriging literature and presents the basic equations involved in kriging.

Chapter III covers the mathematical development of the kriging methods needed specifically for terrain modeling, while Chapter IV covers the implementation of the system used generate terrain models based on the kriging techniques. Chapter V presents the results of this effort and Chapter VI presents conclusions drawn from these results. Appendix A presents further development of the kriging equations from Chapter II. Appendix B presents histograms of various DTED cells, while Appendix C presents variograms generated for these same DTED cells, both in support of Chapter III. Appendix D is the User's Manual and `Unix` man pages for the terrain modeling system. Appendix E describes the `.pts` file format used by the terrain modeling system developed by this thesis effort.

II. *Background on Terrain Modeling and Kriging*

A survey of current literature reveals several methods used to generate polygonal-based terrain models specifically for use in flight simulators and other terrain imaging systems. However, the general field of terrain modeling is active, as people in the computer graphics community are continually developing more advanced techniques for such modeling. This chapter reviews both the popular and the state-of-the-art terrain modeling methods in support of developing a statistically-based method for terrain modeling based upon kriging. *Kriging* is a geostatistical technique traditionally used to estimate ore grades and surfaces from sampled data. Kriging is applied in this thesis to a similar problem, that of estimating terrain elevation values at unsampled locations based on sampled elevation data. As there is little literature on the application of kriging to such analysis and processing of terrain elevation data, this chapter also presents a general overview of the kriging principles to support the development of the kriging methods in Chapter III.

2.1 Terrain Modeling For Flight Simulators

2.1.1 Terrain Model Sources. Most terrain models for flight simulation are based on some form of a terrain elevation database that provides discrete elevation data across specific areas (20)(50)(34)(40). Several organizations, including the Defense Mapping Agency (DMA), provide basic terrain elevation data in digital form (14)(13). DMA provides this data through a product known as Digital Terrain Elevation Data (DTED). DMA DTED provides the basic quantitative terrain data for all military training, planning, and operating systems that require terrain elevation, slope, and/or surface roughness information (1:262). DMA produces two types of DTED, Level 1 and Level 2. Either level provides gridded elevation data partitioned into quadrangles (or cells) of 1 degree latitude by 1 degree longitude; these cells do not overlap whole integer latitudes and longitudes. Within these cells, terrain elevation in meters is provided for specific elevation posts, with the location of these posts being defined as the intersection of the rows and columns of a regular grid overlaying the cells. Level 1 DTED provides elevation data with a minimum post spacing of three arc seconds of latitude or longitude for most cells; Level 2 DTED

Zone	Latitude	Level 1 DTED lat × long	Level 2 DTED lat × long
I	0° - 50° N-S	3 × 3 seconds	1 × 1 seconds
II	50° - 70° N-S	3 × 6 seconds	1 × 2 seconds
III	70° - 75° N-S	3 × 9 seconds	1 × 3 seconds
IV	75° - 80° N-S	3 × 12 seconds	1 × 4 seconds
V	80° - 90° N-S	3 × 18 seconds	1 × 6 seconds

Table 2.1. DMA DTED Post Intervals

provides this data with a one arc second minimum post spacing, but is only available for limited areas of the world. The actual spacing varies according to the the cell's latitude as defined in Table 2.1 (note that all values in seconds are in terms of arc measure). In general, the DTED Level 1 post spacing is roughly 100 meters for any cells, while DTED Level 2 maintains a post spacing of approximately 30 meters. DTED does not take into account the height of any vegetation or other cultural features (1). Additional information on DMA DTED is contained in (14) and (13).

2.1.2 Terrain Modeling Techniques.

2.1.2.1 Simple Polygonal Modeling. As one of the most readily available gridded terrain elevation databases available, DTED is a convenient source of terrain elevation data that can be a basis for a reasonably accurate and realistic terrain model (40)(52)(23)(31)(34). Polygonal models can easily be generated from this data because it is regularly gridded. Triangular polygons can be constructed from such data by connecting the corners of each grid cell and diagonally bisecting the resulting rectangle with a fifth edge (40:6)(6)(34)(42). Since any three points in three-space must be co-planar, the use of triangular polygons also ensures that the resulting polygons are planar; many rendering algorithms require planar polygons in order to render the model quickly (40:3)(23:15).

As discussed in Chapter I, simply connecting the data points provided from a source such as DMA DTED in the fashion described above may create a terrain model much too complicated to maintain a desirable display update rate. The goal of most flight simulators or terrain viewers is to display images at a rate comparable to a motion picture

or real-time animation. Motion pictures use a standard rate of twenty-four frames per second (34:9)(31:14)(20:43); animation is usually done at 30 frames per second (20:43). In general, the faster the display update rate, the more continuous the motion appears (31:14). In most systems, this update rate is most directly affected by the number of polygons that must be rendered for any scene, and the terrain model is usually a major source of such polygons (20:43). A polygonal terrain model built directly from one cell of the DTED could require 2,880,000 triangular polygons, each roughly 100 meters on a side, if modeled in the fashion described above. The density of the polygons in such a model has a direct effect on the display update rate for the system; there is therefore a trade-off between the degree of variation in the terrain model and the speed that the model can be displayed (52:26).

As most flight simulator systems cannot handle the density of the polygonal descriptions obtained directly from DMA DTED data and maintain the desired update rate, this trade-off must be considered and the terrain model must be simplified. Therefore, a polygon budget must be adhered to in order to maintain the desired update rate. A *polygon budget* refers to the number of polygons that can be rendered without degrading the simulated motion of the user over the terrain model (40:13). As different display systems have different polygon budgets, the user must select the appropriate polygon density to keep within their particular polygon budget (40:13). Two methods of creating a terrain model with a specific polygon density are through terrain elevation data filtering and multiple levels of detail.

2.1.2.2 Data Filtering Techniques. Data filtering or reduction techniques refer to any process that can quantify a portion of the elevation data with a single data point. By dividing the entire terrain database into portions and quantifying each portion with only one data point, a less complex terrain model can be produced. One such filtering technique is *sampling* (40)(52)(23:15)(31). This is typically accomplished by selecting every nth datum in the original grid, with n depending on criteria such as the performance characteristics of the target system (as reflected by the polygon budget) and the desires of the designers. This process improves the system's interactive response by increasing the

image update rate, but it also reduces the visual realism of the images generated from this model, as much of the detail present in the original terrain elevation database is lost.

Other approaches to filtering complex terrain data include selecting an elevation for each grid location that is representative of the elevation in that area or interpolating a value for each grid location from the adjacent elevations (40:6). In most cases, the resulting mesh of terrain elevations can still be treated as a regular grid. Depending on the criteria used to select the value for every *n*th datum, this technique can increase the accuracy and the degree of variability of the resulting terrain model without much additional overhead as compared to the strict down-sampling technique reviewed above. However, there is no guarantee that the detail present in the original terrain data is preserved in models generated with these techniques, and there is also no measure of the accuracy of such models.

2.1.2.3 Multiple Levels of Detail. The filtering of data as described above can be used in conjunction with some form of *complexity management* as defined by Veron (50:55). One method of complexity management is to generate several models of each object that is to be rendered by the flight simulator, each with a different level of detail, and to use a model with an appropriate level of detail when rendering the image (40)(20:49)(50)(27:723,733)(45:24). The appropriate model can be selected in many ways. One method of choosing the appropriate model is by the area of the screen that the rendered object will occupy, as the amount of time spent on rendering an object should be proportional to this area (40:2). This implies a less detailed model for objects with smaller rendered images. Another method is to choose the appropriate model based on the distance from the viewer to the object; a more distant object should be rendered from a less-detailed model.

Applying multiple levels of detail also increases the perceived realism of the rendered scenes, as explained by Kennie:

> As an object recedes from a viewer, the object diffuses through several forms of apparent simplification until it eventually disappears from sight. This is due to the limited resolution of the eye's optical system and is influenced by

> atmospheric distortion and the shape and texture of the object... Any image that is attempting to simulate realism in a spatially deep scene should include a mechanism to vary the perceived level of detail of an object based on its distance from the viewer. (27:723)

To apply multiple levels of detail, the terrain model must be divided into sub-cells that can be handled as individual objects by the flight simulator. Based upon the distance from the viewer to an object or sub-cell of terrain, or alternately based upon the screen area that the rendered object or sub-cell occupies, the flight simulator should use an appropriate model for that object or sub-cell. A less detailed model consisting of fewer polygons is used for objects or terrain cells that are so distant from the viewer or that are so small relative to the total screen area that most detail cannot be perceived. A specific level of detail model for a particular object can be provided by either storing multiple versions of the object in a database or by instantiating the object at a selected level of detail with procedural modeling (40:11). Additional memory overhead, disk space and preparation time may be needed to use models with multiple levels of detail, but the result can improve simulator performance and maintain both the interactive responsiveness of the system and the realism of the images rendered (20)(50). However, this technique still provides no measure of accuracy of the resulting terrain model.

2.1.2.4 Other Techniques. Several flight simulator systems use various other techniques to produce acceptable terrain models from terrain elevation databases. Veron uses a Laplacian filter on the original elevation data to detect significant features in terrain; these features dictate areas where greater polygon density is needed to properly model the terrain (50:31). Other techniques analyze the elevation data and select terrain points that are local elevation minimums or maximums. However, either of these techniques can result in a pattern of elevation points that may not be efficiently represented by a regular grid, and a polygonal model is more difficult to produce from such irregular data. Various methods, such as Delauney triangularization, and radial-sweep algorithms (33)(22), have been used to generate a triangular irregular network (TIN) from irregular data. Such methods could provide a basis for building a polygonal terrain description of the irregular terrain data; however, these methods are beyond the scope of this thesis.

2.1.3 General Terrain Modeling. There are several other methods in the current literature that discuss terrain modeling such that the resulting models are visually pleasing. The primary purpose of these methods, however, is not to create realistic and efficient terrain models but to create models that simply appear realistic without regard to complexity. Two of these methods are fractals and erosion models.

Fractal techniques can generate complex detail by *amplifying* a small database of structural or statistical primitives (44)(37)(32). The name *database amplification* refers to the generation of controlled random detail from a fairly sparse description (44). Using this recursive method, a terrain elevation database can be amplified to almost any degree, effectively creating different levels of detail. However, these methods introduce detail through the use of a Gaussian random variable, which may produce a visually realistic image but not necessarily an accurate image (32). Szeliski (46) introduces a method of constraining fractals using splines, forcing the randomness of the fractals to generally follow the path defined by an underlying spline. However, this method does not ensure the accuracy of the interpolated data.

A method of modeling terrain based on the effects of stream erosion has also been proposed, amplifying the terrain elevation database using empirical erosion models from geomorphology (26). A tributary model must be extracted from the elevation database, and by increasing or decreasing the number of tributaries in this model, varying levels of detail may be generated. Although this method generates a realistic terrain model based on the laws of nature that formed the actual terrain, it does not provide a reasonable way to measure the accuracy of any resulting model.

2.1.4 Summary This section has reviewed current literature on various methods of terrain modeling. Popular methods for flight simulator terrain modeling and advanced terrain modeling techniques were covered. Multiple levels of detail models using terrain models based on filtered DTED is the most comprehensive method for modeling terrain for flight simulators. However, there is no technique to finely control the resolutions of the different levels of detail, as may be necessary to achieve maximum performance. Additionally, no method referenced provides any quantitative measure of accuracy of the model

generated based on the original data. A stochastic method of generating specific levels of detail, such as kriging (as described in the next section) would provide both fine control of resolution and a quantit .. e measure of accuracy.

2.2 *Kriging*

Kriging is a geostatistical process for estimating various spatial properties of a regionalized variable. It is a method for estimating and fitting a continuous surface to a set of sample values taken from the true surface of a regionalized variable. Kriging is an optimal interpolation method based on a structural analysis of the sampled data set (10:239). Kriging should be useful in constructing terrain models in that it can provide an optimal estimate of the terrain elevation at any point along with a measure of that estimate's accuracy. However, there has been little use of kriging outside the geostatistical fields, and literature on its application to terrain modeling is limited. Therefore, this background section covers the origin and the basic principles of kriging as described in geostatistical literature.

2.2.1 Origin of Kriging. In order to understand kriging, a basic understanding of the field of geostatistics is necessary. Geostatistics is a term applied to the special branch of applied statistics originally developed by Georges Matheron (12:239). Matheron (29:1246) defines geostatistics as the study of the distribution in space of useful values for mining engineers and geologists; such useful values include grade, thickness, and accumulation. It is also concerned with the practical application of these distributions to problems arising in ore-deposit evaluations.

In 1951, D. G. Krige, a South African mining engineer, advocated the use of a statistical regression technique to estimate ore properties. This technique exploited the spatial correlation of the core samples' grades and the overall block grade while using a simple weighted averaging technique to estimate the ore grade within the block (10:245)(24:563). He proposed that the core samples on the periphery of the unknown site (in its neighborhood) be given equal weight, while all other samples not be considered at all (be given zero weight) (10:245). Although not kriging as we know it today, Krige's methods did

provide the basis he and others later expanded upon to establish the modern geostatistical methods.

In 1962 and 1963, Georges Matheron, a French engineer of the Corps des Mines, published the first complete accounts of the theory of kriging (11:70)(10:246). Matheron also advanced the theory by avocating that weights used in the estimation process should be based upon the covariances within the ore grade as defined by the actual sampled data (10:244). Matheron's work established the term "geostatistics" as meaning "geostatistical ore reserve estimation" (11:70). Although he did not coin the name *kriging* for this method, Matheron did bring it into common usage in Anglo-Saxon mining terminology (10:240). His work lead directly to the technique known today as *block kriging*; this and other variations of kriging are covered later in this chapter.

Since Matheron's original work, kriging has gained wide acceptance in most French mining operations while only slowly reaching the rest of the world. According to David, by 1977 "over two hundred deposits [had] been successfully estimated using this method, including many different types of mineralization" (11:70). The French Atomic Energy Commission, the Italian Uranium Branch, and private uranium mining interests in the United States have successfully used kriging to prospect for uranium deposits; even the oil industry has applied the technique successfully (11:70-71). Many other fields besides geology have successfully used kriging or variations thereof to analyze and estimate spatial properties based on sampled data. Forestry, physics, plant and animal breeding, soil science, crop science, and epidemiology have all found use of this technique in some form (10:249-50)(8:405-6). However, the specific techniques applied in these various areas were often developed independently. A variation of kriging has even been used to predict the movements of enemy aircraft based on known radar measurements during World War II (10:250). More recently, research at the United States Air Force Institute of Technology (AFIT) used kriging to analyze satellite imagery (30), to compress various data sets such as three dimensional Magnetic Resonance Imaging (MRI) data (2), and in the analysis of anthropometric data to help improve the design of flight equipment (19).

2.2.2 Definition of Kriging. According to Cressie,

> Geostatistics... is concerned with a particular class of problems whose data can be modeled according to the stochastic process
>
> $$\{Z(s) : s \in D\}$$
>
> The multivariate datum $Z(s)$ is observed at spatial location s which varies continuously over D, a subset of two- or three-dimensional space R^2 or R^3. Data locations $\{s_i : i = 1, ..., n\}$ are assumed known and... $Z(\cdot)$ is assumed to be real valued. On occasion, this form of stochastic process $[Z(s)]$ has been called a regionalized variable to emphasize its continuous spatial index. (8:406)

The concept of a regionalized variable (ReV) is explained by Davis as follows:

> [A regionalized variable] has properties intermediate between a truly random variable and one completely deterministic. Typical regionalized variables are functions describing natural phenomena that have geographic distributions, such as the elevation of the ground surface, changes of grade within an ore body, or the spontaneous electrical potential measured in a well by a logging tool. Unlike random variables, regionalized variables have continuity from point to point, but the changes in the variable are so complex that they cannot be described by any tractable deterministic function.(12:239)

D. G. Krige defines kriging as "the name given by Matheron to: the multiple regression procedure for arriving at the best linear unbiased [predictor] or best linear weighted moving average [predictor] of the ore grade of an ore block (of any size) by assigning an optimum set of weights to all the available and relevant data inside and outside the ore block" (10:240). Matheron further defines kriging to be the probabilistic process of obtaining the best linear unbiased estimator of an unknown variable (24:563); with "best" meaning having the smallest estimation variance (5:104). Cressie (8:406) defines kriging as a method of predicting the value of a regionalized variable $Z(s_o)$ at an unsampled location s_o based solely upon univariate sample data $\{Z(s_i) : i = 1, ..., n\}$. This definition of kriging allows it to be applied to regionalized variables of any dimension, but for the purposes of this thesis, a two dimensional sample area (spatial data) is assumed. This technique can be extended to a surface fitting technique by predicting $Z(s_o)$ at all unsampled locations s_o in order to produce an estimated surface. It is important to note that kriging is not a filtering or estimation technique where the goal is to predict a noiseless version of $Z(s_o)$ (10:241); neither is it a database amplification technique as are fractal methods (44)(26)(37)(32).

Many other estimation techniques, including forms of interpolation, trend analysis, and moving averages, would also estimate $Z(s_o)$ with varying degrees of accuracy (36)(41). However, a tremendous amount of support exists for the practical application of kriging in problems such as surface estimation due to its widespread use in mining. A. G. Journel, a leading proponent of kriging, states "there is a long, hard way between a concept... and its implementation and routine application... The main contribution of geostatistics has been and still is *implementation*, an essential follow-up step much too often forsaken by theoreticians" (25). Other advantages of kriging include the fact that kriging is an exact interpolator; that is, the estimate provided by kriging at any known sample point is that sampled value exactly. Kriging also provides an inherent measure of the error or uncertainty of the contoured surface in the form of an error variance (12:383)(9:199).

Although kriging originally involved linear predictors, its name has since been used to refer to other methods that provide optimal nonlinear spatial predictors of a regionalized variable (10:240). These techniques are beyond the scope of this research.

2.2.3 Kriging Equations. The following brief development of *ordinary kriging* is adapted mostly from Journel (25) and Davis (12) (the notation has been changed to be consistent with other sources). This description also provides the basis for presenting other forms of kriging later in this section. A more complete development of these equations is provided in Appendix A.

The intent of kriging is to predict the value of an unknown property of a regionalized variable at a particular location using a weighted sum of actual sampled values. Kriging estimates the unknown quantity $Z(s_0)$ associated with an unsampled location s_0 using an estimator $Z^*(s_0)$ that is a weighted average of n sample values $Z(s_i), i = 1, ..., n$, from locations s_i where $Z(s_i)$ is known:

$$\begin{aligned} Z^*(s_0) &= w_1 Z(s_1) + w_2 Z(s_2) + w_3 Z(s_3) + \cdots + w_n Z(s_n) \\ &= \sum_{i=1}^{n} w_i Z(s_i) \end{aligned} \tag{2.1}$$

The weights w_i are selected in such a way that $Z^*(s_0)$ yields the best possible unbiased

estimator for $Z(s_0)$. As stated earlier, kriging defines the "best" estimator as the estimator with the minimum error variance (or estimation variance (12:238)(5:104) or mean-squared prediction error (10:241)). The general error variance σ_E^2 at any estimated point s_0 is given by Eq (2.2):

$$\begin{aligned} \sigma_E^2 &= Var[Z(s_0) - Z^*(s_0)] \\ &= \sigma^2 - 2\sum_{i=1}^{n} w_i\sigma_{i0} + \sum_{i=1}^{n}\sum_{j=1}^{n} w_i w_j \sigma_{ij} \end{aligned} \tag{2.2}$$

where $\sigma_{ij} = Cov[s_i, s_j]$, the covariance between points s_i and s_j; $\sigma_{i0} = Cov[s_i, s_0]$, the covariance between points s_i and the point being estimated s_0; and $\sigma^2 = Var[Z]$, the variance of the data set.

The unbiasedness condition requires that, on average, the estimates should be equal to the real value, rather than systematically higher or lower (11:238). This is ensured by requiring the weights to sum to one:

$$\sum_{i=1}^{n} w_i = 1 \tag{2.3}$$

The weights that minimize the error variance can be determined by taking the differential of the error variance with respect to the weights and setting each of these partial derivatives to zero:

$$\frac{\partial(\sigma_E^2)}{\partial w_i} = 0, \quad i = 1, \ldots, n \tag{2.4}$$

This results in a system of n equations with n unknowns. The unbiasedness condition given in Eq (2.3) requires the addition of another equation to the system, and the LaGrange multiplier λ is added to sufficiently constrain the system (12:385). The entire system of equations is shown below:

$$
\begin{array}{ccccccccccc}
w_1\sigma_{11} & + & w_2\sigma_{12} & + & \cdots & + & w_n\sigma_{1n} & + & \lambda & = & \sigma_{10} \\
w_1\sigma_{21} & + & w_2\sigma_{22} & + & \cdots & + & w_n\sigma_{2n} & + & \lambda & = & \sigma_{20} \\
w_1\sigma_{31} & + & w_2\sigma_{32} & + & \cdots & + & w_n\sigma_{3n} & + & \lambda & = & \sigma_{30} \\
\cdots & + & \cdots & + & \cdots & + & \cdots & + & \cdots & = & \cdots \\
w_1\sigma_{n1} & + & w_2\sigma_{n2} & + & \cdots & + & w_n\sigma_{nn} & + & \lambda & = & \sigma_{n0} \\
w_1 & + & w_2 & + & \cdots & + & w_n & & & = & 1
\end{array}
\tag{2.5}
$$

In matrix form, these equations can be represented as:

$$
Aw = B \tag{2.6}
$$

where A, w, and B refer to the following matrices:

$$
A = \begin{bmatrix}
\sigma_{11} & \sigma_{12} & \cdots & \sigma_{1n} & 1 \\
\sigma_{21} & \sigma_{22} & \cdots & \sigma_{2n} & 1 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
\sigma_{n1} & \sigma_{n2} & \cdots & \sigma_{nn} & 1 \\
1 & 1 & \cdots & 1 & 0
\end{bmatrix}
\tag{2.7}
$$

$$
w^T = \begin{bmatrix} w_1 & w_2 & \cdots & w_n & \lambda \end{bmatrix} \tag{2.8}
$$

$$
B^T = \begin{bmatrix} \sigma_{10} & \sigma_{20} & \cdots & \sigma_{n0} & 1 \end{bmatrix} \tag{2.9}
$$

To obtain the weights that minimize the error variance, this matrix equation is solved for w:

$$
w = A^{-1}B \tag{2.10}
$$

This can be done once the covariances are known; a function called the semivariance or semivariogram ultimately provides these covariances (see Section 2.2.7 below). The weights

obtained by solving this matrix equation can then be substituted into the original weighted sum predictor (Eq (2.1)), yielding an estimate of $Z(s_0)$ with minimum error variance; the value of the LaGrange multiplier λ can be ignored. The exact error variance at s_0 can also be computed, using Eq (2.2), to provide a measure of the accuracy of the estimate (12). In matrix form, this equation is:

$$\sigma_E^2 = \sigma^2 - w^T B \tag{2.11}$$

where σ^2 is the variance of the data set.

Since the final linear estimator $Z^*(s_0)$ is unbiased and considered a best estimate with respect to the error variance, the estimator is referred to as a best linear unbiased estimator, or *blue*, by statisticians (11:237). These equations are also known as the *constrained normal* system of equations or the equations for *linear regression under constraints* (25:15-16), but will simply be referred to here as the *ordinary kriging equations*.

2.2.4 Neighborhoods The system of equations given in 2.5 show that the weights w_i depend on the covariance between the locations of the point s_i and the point being estimated s_0, not on the values of the function Z at these locations. For a regionalized variable, the covariance between locations varies inversely with their distance apart; it therefore follows that nearby samples should have greater weight than farther samples, with distant samples receiving no weight. This shows that the concept of neighborhoods or zones of influence is inherent in kriging (11:76). Davis defines a neighborhood in the following fashion:

> For some arbitrary point in space, we can imagine the neighborhood as a symmetrical interval about the point... the neighborhood is defined as a convenient but arbitrary interval within which we are reasonably confident that all locations are related to one another. (12:240,244)

Generally, sample data from points outside the of neighborhood are not included in the kriging system of equations. In practice, the size of this neighborhood must be assumed

THIS
PAGE
IS
MISSING
IN
ORIGINAL
DOCUMENT

2.2.6 Kriging with a Trend. As stated in Section 2.2.5 above, data can violate the stationarity assumption by either exhibiting global trend or local drift. A regionalized variable with global trend can be modeled by Eq (2.12):

$$Z(s_p) = M_{\text{global}}(s_p) + R(s_p) \tag{2.12}$$

where $Z(s_p)$ represents the regionalized variable under consideration, $M_{\text{global}}(s_p)$ is the global trend, and $R(s_p)$ represents the residuals that are *globally* stationary. $M_{\text{global}}(s_p)$ is usually a first or second order polynomial as shown in Eqs (2.13) and (2.14):

$$M_{\text{global}}(s_p) = a_0 + a_1X(s_p) + a_2Y(s_p) \tag{2.13}$$

$$M_{\text{global}}(s_p) = a_0 + a_1X(s_p) + a_2Y(s_p) + a_3X(s_p)^2 + a_4Y(s_p)^2 + a_5X(s_p)Y(s_p) \tag{2.14}$$

where a_i are unknown global trend coefficients and $X(s_p)$ and $Y(s_p)$ are the x and y coordinates of the point s_p. Global stationarity can be ensured in discrete sampled data by using a method such as least squares regression to fit a polynomial $M_{\text{global}}(s_p)$ to the sampled data and subtracting this polynomial from the sampled data $Z(s_p)'$. Depending on the polynomial used, the remaining residuals $R(s_p)'$ should exhibit a greater degree of global stationary (41).

The removal of local drift, however, is easiest performed concurrent with the kriging. The variation of kriging known as *universal kriging* can be used to account for not only the stochastic behavior of the regionalized variable but also for the underlying local drift. Universal kriging is an expansion of ordinary kriging where equations representing a polynomial drift are added to the system of equations and the coefficients for these polynomial components are calculated concurrent with the weights (12:393). Since universal kriging is used in this thesis, its justification and development follows.

The local drift can be defined as drift localized to the neighborhood around the location being estimated. Ordinary kriging estimates the value of an unknown point based upon the values of other points in its neighborhood; these same points can be used to solve

for local drift. Davis defines local drift $M_{\text{local}}(s_p)$ at any point s_p as a first-order (Eq (2.15)) or second-order (Eq (2.16)) polynomial:

$$M_{\text{local}}(s_p) = \alpha_0 + \alpha_1 X(s_p) + \alpha_2 Y(s_p) \tag{2.15}$$

$$M_{\text{local}}(s_p) = \alpha_0 + \alpha_1 X(s_p) + \alpha_2 Y(s_p) + \alpha_3 X(s_p)^2 + \alpha_4 Y(s_p)^2 + \alpha_5 X(s_p) Y(s_p) \tag{2.16}$$

where α_i are unknown local drift coefficients and $X(s_p)$ and $Y(s_p)$ are the x and y coordinates of the point s_p (12:394). One of these expressions is incorporated into the system of simultaneous equations shown in Eq (2.5) used to find the kriging weights. The first or second order polynomial surface is fitted to the points local to the estimated point, solving for the local drift coefficients specific to that area. By solving for the local drift simultaneously with the kriging weights, these weights includes the effects of the drift (12:394); the coefficients α_i are treated as LaGrange multipliers and can be ignored. Once the weights are determined, the estimate $Z(s_0)$ is calculated using Eq (2.1) as with ordinary kriging.

The matrix equations for universal kriging with linear or first-order trend are defined as follows (quadratic or second-order trend is incorporated in a similar fashion):

$$A = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1n} & 1 & X_{s_1} & Y_{s_1} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2n} & 1 & X_{s_2} & Y_{s_2} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ \sigma_{n1} & \sigma_{n2} & \cdots & \sigma_{nn} & 1 & X_{s_n} & Y_{s_n} \\ 1 & 1 & \cdots & 1 & 0 & 0 & 0 \\ X_{s_1} & X_{s_2} & \cdots & X_{s_n} & 0 & 0 & 0 \\ Y_{s_1} & Y_{s_2} & \cdots & Y_{s_n} & 0 & 0 & 0 \end{bmatrix} \tag{2.17}$$

$$w^T = \begin{bmatrix} w_1 & w_2 & \cdots & w_n & \lambda & \alpha_1 & \alpha_2 \end{bmatrix} \tag{2.18}$$

$$B^T = \begin{bmatrix} \sigma_{10} & \sigma_{20} & \cdots & \sigma_{n0} & 1 & X_{s_0} & Y_{s_0} \end{bmatrix} \tag{2.19}$$

The matrix equation $w = A^{-1}B$ is solved as before to determine the weights.

2.2.7 Structural Analysis. The covariances needed to complete either the ordinary or universal kriging system of equations can be realized using a function called a semivariance or semivariogram. The *semivariogram* γ and the *variogram* (which is simply 2γ) depicts a directional semivariance or the rate of change of a regionalized variable along a specific orientation (12:240). As the terms variogram and semivariogram refer to the same basic function, further development refers to both forms simply as the variogram.

Determining this variogram is a necessary first step to kriging. According to Yakowitz and Szidarovszky:

> The kriging method is composed of two activities, (i) inferring the variogram from the data and (ii) assuming that the inferred variogram is indeed exact, providing a best linear unbiased estimator and associated error variance. (51)

Therefore, the variogram is the product of a structural analysis step that must be performed on the regionalized variable before the value at any unsampled point can be estimated.

In general, the variogram is a function that provides the "average difference squared" between data a given distance apart in a given direction (28). It is defined by Eq (2.20):

$$2\gamma(s_1, s_2) = Var[Z(s_1) - Z(s_2)] \tag{2.20}$$

For a true stationary regionalized variable with no trend, the differences of variables lagged h apart vary in a way that depends only on h and not on the spatial positions or the actual values of the variables (9). This allows the variogram to be depicted as $\gamma(h)$, a function of h, where h refers to the distance between two points s_1 and s_2 in the regionalized variable (41). This form of the variogram is defined by

$$2\gamma(h) = Var[Z(s+h) - Z(s)] \qquad s,\ s+h \in D \tag{2.21}$$

The following intuitive description of the variogram is adapted from Davis (12:240). If the lag Δh between two sampled points is small, the points being compared tend to

be very similar and the semivariance is a small value. As the distance Δh is increased, the points being compared become less closely related and their differences become larger, resulting in larger values of $\gamma(h)$. Eventually, the points being compared are so far apart that they are not related to each other, and their squared differences become equal in magnitude to the variance around the average value.

In the practical application of kriging, the variogram $\gamma(h)$ is estimated by using the following discrete function $\gamma'(h)$, known as the experimental variogram:

$$\gamma'(h) = \frac{1}{2|N(h)|} \left\{ \sum_{i=1}^{N(h)} [Z(s_i) - Z(s_{i+h})]^2 \right\} \tag{2.22}$$

where the sum is taken over $N(h) = \{(s_i, s_j) : |s_i - s_j| = h\}$. In other words, the sum is taken over every pair of s_i and s_j in the regionalized variable that are h units apart; $|N(h)|$ is the number of such pairs of points (41). The experimental variogram is, in effect, a measure of the degree of spatial dependence between samples (12). Figure 2.1 shows a sample experimental variogram.

As the experimental variogram function γ' is not continuous, the actual equation used for the variogram γ is determined by fitting a continuous function (or model) through the discrete points of this experimental variogram γ'. Davis, among others, does not view variogram model fitting as an exact science; Davis states that such model fitting "is to a certain extent an art, requiring experience, patience, and sometimes luck" (12:245). However, Cressie references work where several methods of variogram-model fitting and parameter estimation were performed and it was found that the weighted least squares approach to fitting a model to an experimental variogram "usually performs well and never does poorly" (9:198).

A weighted least squares method can be used to fit a continuous function, or variogram model, to the experimental variogram. Davis provides three typical variogram models used extensively in geostatistics: the linear model, the De Wijsian model, and the spherical model (11). The functions representing these models are given in Eqs (2.23), (2.24),

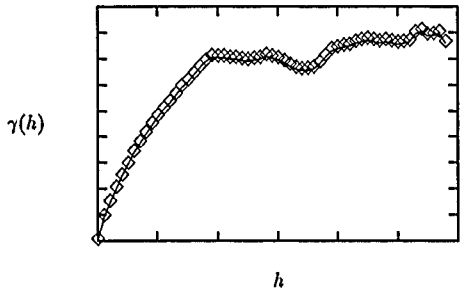


Figure 2.1. Typical Experimental Variogram Generated From Terrain Data

and (2.25) and are shown in Figures 2.2, 2.3, and 2.4.

$$\text{Linear Model}: \quad \gamma(h) = ah + b \tag{2.23}$$

$$\text{De Wijsian Model}: \quad \gamma(h) = a \ln h + b \tag{2.24}$$

$$\text{Spherical Model}: \quad \gamma(h) = \begin{cases} C\left(\frac{3}{2}\frac{h}{a} - \frac{1}{2}\frac{h^3}{a^3}\right) + C_0 & \text{if } h < a \\ C + C_0 & \text{if } h \geq a \\ 0 & \text{if } h = 0 \end{cases} \tag{2.25}$$

The spherical model has several properties that are desirable in the field of goestatistics and is therefore the most commonly used variogram model (11:102). It is defined by three parameters: a, C_0, and C. The first parameter a is defined as the range. This parameter represents the distance where the variogram becomes maximum, or the covariance becomes minimum; it defines a range of influence and can be used to define the size of the kriging neighborhood (19). The second parameter, C_0, represents a combination of sampling error and small scale variability of the data (variability exhibited between data at a finer scale than the sampling interval) (28). It is commonly referred to as the *nugget effect* and is exhibited in the model at the point the variogram model intersects the $\gamma(h)$ axis (28). By definition the variogram model is zero at $h = 0$, indicating that any two data $h = 0$ units apart are completely correlated; this ensures an *exact interpolator* as defined

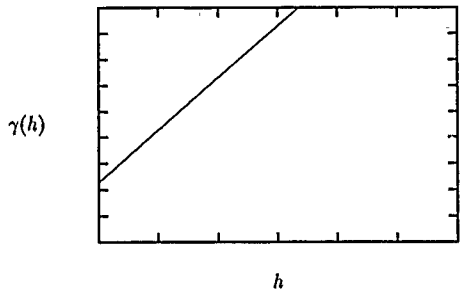


Figure 2.2. Sample Linear Model Variogram

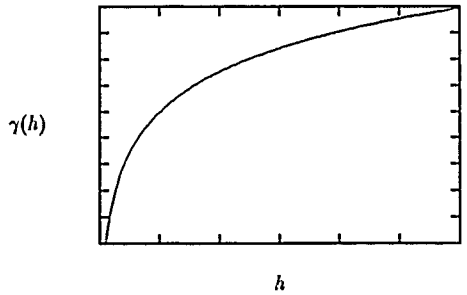


Figure 2.3. Sample De Wijsian Model Variogram

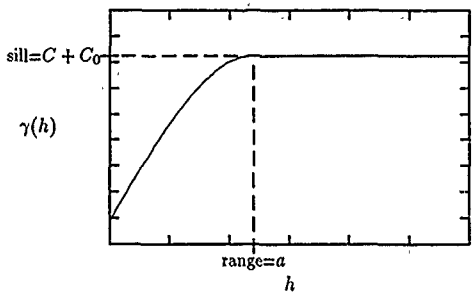


Figure 2.4. Sample Spherical Model Variogram

above. The nugget effect arises because the regionalized variable is so erratic over a very short distance that the semivariogram goes from zero to the level of the nugget effect in a distance less than the sampling interval (12:246). The second and third parameters together, $C + C_0$, represents the *sill* of the data, which is equal to the variance of the data. Additionally, the slope of the model between $h = 0$ and $h = a$ is an indication of the continuity of the data within the zone of influence (28).

The covariance σ_{ij} is similar to the variogram $\gamma(h)$ in that the covariance between two points s_i and s_j lagged h apart depend only on h and not on their spatial positions (9). The covariance σ_{ij} can therefore written as a function of the variogram γ, which is in turn, a function of the distance h:

$$\begin{aligned} \sigma_{ij} &= \sigma(\gamma(h)), \quad h = |s_i - s_j| \\ &= \gamma(\infty) - \gamma(h) \\ &= \sigma^2 - \gamma(h) \end{aligned}$$

where σ^2 is the variance of the data set. If the spherical variogram model is used, then $\sigma^2 = C + C_0$, and the equation for the covariances becomes:

$$\sigma_{ij} = C + C_0 - \gamma(h) \tag{2.26}$$

The covariances needed to estimate $Z(s_0)$ are easily calculated using this equation.

It is possible to write the kriging system of equations by directly using the variogram instead of covariances, and many references develop kriging in this fashion. This yields a system of equations of the following form:

$$\begin{aligned} &\textstyle\sum_j w_j \gamma(h_{ij}) = \gamma(h_{i0}) \\ &\textstyle\sum_i w_i = 1 \end{aligned} \tag{2.27}$$

where h_{ij} represents the distance between s_i and s_j. However, since $\gamma(h_{ii}) = 0$ for any position s_i, this system yields a matrix form with a zero diagonal. Such a matrix is not easily inverted and therefore produces a system of equations that are much more difficult to solve than the equations developed using covariances (11:241-2).

As stated above, variogram generation, like kriging itself, assumes that the regionalized variable is stationary. If the sample data does not exhibit stationarity, this trend must be removed before the variogram is computed (12:244). If the trend is global, it can be removed much like global trend is removed before kriging (see Section 2.2.6 above). However, Davis advocates that the removal of local drift is a circular problem (12:244-5). He presents an iterative trial and error method of assuming a variogram and using universal kriging to remove this local drift. A variogram is generated on the resulting residuals, and if it matches the assumed variogram, the trend has been successfully removed. If the new variogram does not match the assumed variogram, the assumed variogram must be modified and the process is repeated. Davis presents no heuristic on assuming an initial variogram or on modifying it if it does not match.

In addition to the stationarity requirement, the regionalized variable must also be isotropic. There are two forms of anisotropy: *zonal* (or *stratified*) and *geometric* (or *affine*) (11:134). These forms of anisotropy are evident when directional experimental

variograms generated on the same data set are dissimilar. Directional experimental variograms are generated by only considering the differences in values between pairs of points that lie in a specific direction from each other; this direction is known as the *regularization angle.* If these directional experimental variograms are modeled with spherical models, zonal anisotropy can be characterized by different sills in the directional variograms (28:56); in other words, the data exhibits different variances in different directions. This is an indication of zones or areas that have differing sources of variation (11:135). Geometric anisotropy is characterized by different ranges but similar sills in the directional variograms. This is an indication that the value of the regionalized variable varies more quickly in one direction than in the other and that the size of the zone of influence differs in different directions (11:68).

Since kriging uses one variogram for all directions, anisotropy must be taken into account. Zonal anisotropy is difficult to handle, but geometric anisotropy can be compensated for with an anisotropic ratio k. It is equivalent to a change in the distance unit on one axis (11:135). One of the variogram models is scaled by k such that it matches the other. The kriged results are inversely scaled to provide correctly spaced results. If a spherical variogram model is used, the resulting equations for the variogram are:

$$\begin{aligned} \gamma_1(h) &= C\left(\frac{3}{2}\frac{h}{a} - \frac{1}{2}\frac{h^3}{a^3}\right) + C_0 \\ \gamma_2(h) &= C\left(\frac{3k}{2}\frac{h}{a} - \frac{1}{2}\frac{k^3h^3}{a^3}\right) + C_0 \end{aligned} \tag{2.28}$$

Clark suggests that semivariograms ought to be constructed in at least two directions to check for anisotropy (5:38).

2.2.8 Other Forms of Kriging. There are several forms of kriging besides ordinary and universal kriging. *Simple kriging* is the most basic form of kriging. It is similar to ordinary kriging with the exception that the mean μ is assumed known. The development of the simple kriging equations is similar to the development of the ordinary kriging equations presented in Eq (2.5). The resulting equations are known as as the *normal* or *linear regression* equations, or for our purposes, the *simple kriging equations* (25:11).

The terms *point, punctual,* and *block kriging* are simply variations of other forms of kriging; the names refer to the type of estimate being made. *Point* or *punctual kriging* uses the system of equations resulting from any of the above forms of kriging to estimate the value of the regionalized variable at a specific point (11:240,242), while *block kriging* uses the same equations to estimate the average value of the regionalized variable over a block (11:240). The only difference in the systems of equations used for each of these is with the covariances; block kriging uses covariances between blocks while punctual kriging uses covariances between points. Block kriging is the method used most often in geostatistics; however, this thesis uses punctual kriging.

When the distribution of the regionalized variable is not normal, some form of preprocessing and postprocessing must be performed on the sample values used in the kriging equations. *Log-normal kriging* handles regionalized variables that are log-normally distributed by taking the log of the sampled values before kriging is performed and taking the inverse log of the resulting estimates to obtain the true estimate (41) (5:34) (21). This is a common form of kriging as many ore distributions are log-normal (11:11).

Dual kriging is a variation of kriging in which the kriging weights are not calculated repeatedly for each estimation. As the weight applied to a known elevation value in the kriging weighted sum is directly related to the distance h from its position to the position being estimated, a function can be determined during the structural analysis phase that models the kriging weights as a function of this distance h. This function should provide the weights much quicker than the system of equations outlined above (41).

Other forms of kriging include *disjunctive kriging,* and *cokriging.* As these methods are beyond the scope of this thesis, the reader is referred to Henley (21) for more information.

2.2.9 Summary This section presented an overview of kriging, a geostatistical estimation technique that provides a best, linear, unbiased estimator for a property of a regionalized variable at an unsampled point. The estimation is made using a weighted sum of the known sample values; the weights are based upon covariances of the actual sampled data as determined during structural analysis. Universal kriging in particular was

presented, as this is the variation of kriging used in this thesis.

2.3 Summary

This chapter covered several topics involving the current state of the art in terrain modeling for flight simulators and other real-time terrain imaging systems. An abundance of literature exists in this area, but the techniques seem to be very similar. Methods of modeling terrain surfaces for other applications were also reviewed. As a statistical method for modeling terrain is desired, this chapter also reviewed the geostatistical estimation technique known as kriging. Since little literature exists on this topic outside the field of geostatistics, this review concentrated on the description of kriging as extracted from geostatistical literature.

III. Development of the Kriging Methodology for Terrain Modeling

As shown by the last chapter, there is little background material on kriging as applied to estimating terrain elevations. Since one of the objectives of this thesis is to show the applicability of kriging as a technique to reduce dense terrain elevation data in order to generate accurate and realistic terrain models, this chapter first shows how the kriging methodology is applicable to simple terrain elevation estimation. The methods of variogram modeling and kriging as applied to terrain elevation estimation are also covered. These methods and the resulting tools that incorporate these methods were developed in conjunction with Brodkin (2) and McGee (30).

3.1 Applicability of Kriging to Terrain Elevation Estimation

Kriging, as presented in Chapter II, is a method of estimating values of a regionalized variable at non-sampled locations. In essence, it is an stochastic interpolation technique that estimates values of spatial properties at unsampled locations based on values of these properties at sampled locations in its neighborhood. Davis describes the elevation of the ground surface, along with other natural phenomena with geographic distributions, as a regionalized variable for the purpose of kriging (12:239). Other stochastic interpolation methods exist that may provide reasonable terrain elevation estimates, but most other such techniques do not provide a measure of error such as an error variance; neither are they exact estimators (41). Additionally, considering the abundance of literature that exists on the practical application of kriging to problems such as surface estimation (as described in Chapter II), this choice seems well justified.

Because of the way that DTED is structured into files containing terrain elevation over $1° \times 1°$ cells, this full cell or some smaller sub-cell resulting from the division of this cell is the most convenient block of data to use while kriging. Therefore, a set of DTED cells, referred to as the *DTED Evaluation Cells*, are used to support the evaluation and development of the kriging methodology in this chapter; these cells are listed in Table 3.1 (Note that this is not the same set of DTED Test Cells used in Chapter I to demonstrate the implementation of the terrain modeling systems).

34°N×106°W	34°N×107°W	35°N×106°W	35°N×107°W	36°N×106°W
36°N×107°W	37°N×106°W	37°N×107°W	43°N×104°W	43°N×105°W
44°N×104°W	44°N×105°W	36°N×118°W	36°N×119°W	36°N×120°W
36°N×121°W	36°N×122°W	36°N×123°W	37°N×118°W	37°N×119°W
37°N×120°W	37°N×121°W	37°N×122°W	37°N×123°W	37°N×124°W
39°N×84°W	39°N×85°W	38°N×104°W	38°N×105°W	38°N×106°W
38°N×107°W	38°N×108°W	39°N×104°W	39°N×105°W	39°N×106°W
39°N×107°W	39°N×108°W	32°N×80°W	33°N×78°W	33°N×79°W
33°N×80°W	34°N×77°W	34°N×78°W	34°N×79°W	34°N×80°W
35°N×76°W	35°N×77°W	35°N×78°W	35°N×79°W	35°N×80°W
35°N×81°W	35°N×82°W	35°N×83°W	35°N×84°W	35°N×85°W
36°N×76°W	36°N×77°W	36°N×78°W	36°N×79°W	36°N×80°W
36°N×81°W	36°N×82°W	36°N×83°W	36°N×84°W	36°N×85°W
43°N×110°W	44°N×110°W	44°N×111°W	34°N×68°E	34°N×69°E
35°N×68°E	35°N×69°E	36°N×74°E	36°N×75°E	37°N×35°E
37°N×36°E	37°N×74°E	37°N×75°E	38°N×35°E	38°N×36°E
38°N×127°E	38°N×128°E			

Table 3.1. DTED 1°×1° Evaluation Cells (Latitude & Longitude is of Southwest Corner)

Kriging assumes that the elevation data is normally distributed in order that analytical statements can be made about the estimate and the error variance (11:116). As this thesis explores the possibility of using the error variance as a measure of accuracy, the first step is to determine if terrain elevation data is normally distributed. Appendix B shows a series of histograms from the set of DTED Evaluation Cells; these histograms plot terrain elevation against the frequency of that particular elevation on both normal and log-normal scales. Frederiksen states that most terrain elevation data is log-normally distributed (17). Many of the histograms in Appendix B appear log-normally distributed, but several could be considered normally distributed. Most DTED cells that are not normally or log-normally distributed contain a large body of water; DMA codes the elevations over a body of water as a constant, skewing the distribution curve, especially when this body of water lies at the lowest elevation in the cell (as would be the case in most cells that contain part of the ocean). The removal of all elevation data pertaining to the body of water would not solve this problem, as an disproportionally large proportion of the remaining elevation data would then correspond to the shoreline and lie just above the elevation of the body of water. This thesis effort ignores the unusual cases and assumes that all cells

are normally distributed, with the understanding that estimates on or near bodies of water are less optimal than estimates made elsewhere.

Stationarity of the elevation data can be assured by removing all global and local trend. The removal of global trend can be handled using a least squares method of fitting a second-order polynomial to the terrain surface; the subtraction of this polynomial from the original sampled terrain data should extract a majority of the global trend. The remaining residuals should exhibit a greater degree of global stationarity than the original data. This trend can easily be replaced into any kriged estimates that are generated. As discussed in Chapter II, local trend can be handled during the actual estimation process using universal kriging. However, considering the difficulty of removing local trend before universal kriging is performed, this effort ignores local trend while generating variogram models.

3.2 Variogram Modeling

Directional experimental variograms of specific DTED cells, as described in Chapter II, can be generated using Eq (2.22). This is easily accomplished if the data is gridded and the direction (or *regularization angle*) ϕ that the variogram is generated is orthogonal to the grid: Eq(2.22) is simply applied using every pair of points that lie in the specified direction ϕ on the grid. If either one of these conditions are not met, a *semi-inclusion angle* ψ must also be specified, as shown in Figure 3.1. The data must then be exhaustively searched; a particular pair of points is included in the calculation of the directional experimental variogram if the angle α of a vector between the pair of points lies between $\phi - \psi$ and $\phi + \psi$. In other words, use every pair of points where the following condition is true:

$$\left| \frac{dy}{d} \cos\phi + \frac{dx}{d} \sin\phi \right| \geq \cos\psi \tag{3.1}$$

where dx is the x component of distance between pair of points, dy is the y component of distance between pair of points, and $d = \sqrt{dx^2 + dy^2}$. Clark (5:38) states that these experimental variograms should to be generated in at least two directions to check for anisotropy. Variograms in the north-south and the east-west directions are easiest to generate if the data is gridded. Methods of identifying and handling anisotropy based on

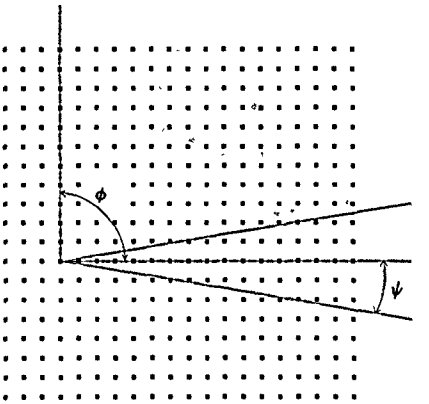


Figure 3.1. Variogram Regularization Angle ϕ and Semi-Inclusion Angle ψ

these variograms are discussed in Section 3.5 below.

Another consideration when computing experimental variograms is the *stepsize*. Since the sampled data that is used to generate this variogram is not continuous (and may not even be regularly spaced), a stepsize Δh needs to be defined such that every pair of points separated by d, $h \leq d < h + \Delta h$, is considered as separated by h for the purpose of variogram generation; h is a multiple of the stepsize Δh. This produces an experimental variogram with datapoints separated by Δh. This stepsize should be large enough such that the value of the experimental variogram for any specific h is based upon a significant number of pairs of samples.

The directional experimental variograms should be generated from the residual data that is obtained by removing the global trend from the DTED cell as described in Section 3.1 above. Clark also (5:14) states that these experimental variograms should be generated for only half of the total sampled extent. This is because the reliability of the experimental variogram diminishes as the lag h between the sampled pairs increases, for

there are fewer pairs of samples separated by such large distances.

Appendix C shows directional experimental variograms that were generated from both the original and residual forms of the DTED Evaluation Cells; both north-south and east-west directional variograms are shown. Variogram models must next be fitted to these experimental variograms. Many consider the fitting of a model to an experimental variogram a complex, trial-and-error process (12:245). However, Robinson (41) states that kriging is fairly robust with respect to the variogram, as a reasonable estimate can be made with a grossly inaccurate variogram.

The experimental variograms of the residuals from Appendix C can be compared with the sample linear, DeWijsian, and spherical variogram models depicted in Chapter II (Figures 2.4, 2.5, and 2.6 respectively). Although several of these experimental variograms are somewhat erratic, it is seen that most variograms for the DTED cells can be modeled using the spherical model. The spherical variogram models shown with the experimental variograms in Appendix C were fitted to half of the total sampled extent using the least squares fitting technique. However, this algorithmic method of fitting a variogram model to the experimental variogram does not always provide an adequate model for kriging. The major problem is that the spherical variogram nugget may be inaccurate, perhaps even negative. The case of a negative nugget must be recognized and handled, perhaps by resetting the nugget to zero. Obviously this method is not perfect, but it does well in most cases, and the use of a single variogram model and an algorithmic variogram model fitting technique facilitates the automation of the kriging estimations.

3.3 Kriging

As the estimation of the terrain elevation at specific points is desired from the terrain modeling system, the method of punctual kriging should be employed. The ultimate output should be a grid of terrain elevations covering the same area as the DTED from which they were extracted; however, points on this new grid may or may not coincide with points on the original DTED grid. Terrain elevation estimation at these points can be done using the kriging methodology described in Chapter II; however, a few of the issues specific to this effort are addressed below.

As stated in Chapter II, only sampled points within a neighborhood around the point being estimated are used in the weighted sum of Eq (2.1) to produce an estimate; all other sampled points are given zero weight. Davis (12:393) states that the optimum number of control points for a neighborhood is determined by the semivariogram and the spatial pattern of the points. The range of the spherical variogram model can be used to define the radius of the neighborhood. Such a neighborhood is adequate for most positions that may be estimated within a DTED cell. Positions near the boundary of the cell do not have a complete neighborhood as defined above, and therefore the estimates made at these positions are not as accurate as other estimates using complete neighborhoods. Extrapolation of terrain elevations beyond the boundary of the cell can also introduce gross errors (12:393).

The first step in kriging is to determine the sampled points in the neighborhood of the point to be estimated. This can be simply achieved by organizing the data into buckets of arbitrary size, with each bucket representing a small area on the sampled surface. An easy way to determine which control points lie in the neighborhood of a specific point to be estimated is to search the buckets within a square region of width = 2×range centered around that particular point. By using covariances instead of semivariances to perform the estimation (as described in Chapter II), a system of equations can be set up that is solved for the unknown weights using matrix algebra. Matrix inversion can be accomplished using the method of LU decomposition (38). The weights obtained by using LU decomposition and backsubstitution can be used to estimate the value at the unknown point directly using Eq (2.1).

Two potential problems are apparent with the kriging estimation process as described above. The first concerns matrix inversion using LU decomposition and backsubstitution. This matrix inversion technique can be unstable on matrices with over approximately 250 rows and columns and on matrices with zero diagonals (41). However, it is expected that most estimated points will have far more than 250 sampled elevation posts in their neighborhoods, and therefore the matrices used during kriging will have far more than 250 rows and columns. Also, LU decomposition and backsubstitution is an $O(n^2)$ algorithm, meaning that kriging elevation estimations over a large regular grid from dense terrain

elevation data will be very time intensive. The second potential problem is associated with the density of DTED data. For a large area, the resulting volume of terrain elevation data used to generate kriged estimates may be overwhelming. Methods should be implemented that allow the volume of data to be limited and the neighborhood sizes to be constrained so that the estimation process is not computationally intractable. This would cause the kriged estimates to be less optimal. However, some evidence of kriging's applicability to terrain modeling can be determined form these less than optimal estimates.

A by-product of kriging is an error variance for every position estimated. This error variance, as described in Chapter II, is a possible measure of accuracy for any resulting terrain model. The error variance is the measure of one standard deviation for the particular estimate it is associated with. Using this value, one can determine a confidence interval for the estimate. For example, if the estimate for location s_0 is $Z^*(s_0)$ and the error variance is $\sigma_E^2(s_0)$, then a 99% confidence interval for this estimate is $Z^*(s_0) \pm 6\sigma_E(s_0)$, where $\sigma_E(s_0) = \sqrt{\sigma_E^2(s_0)}$ Similar statements may be made concerning the estimated terrain model as a whole using the maximum error variance.

3.4 Error Variance Minimization

Given an initial set of sampled positions from a regionalized variable and a set of potential samples to add to this set, Szidarovszky (47)(4) describes a procedure that uses kriging to select the next k positions to sample such that the maximum error variance of the resulting set of points is not greater than if any other k positions were sampled. Of greater interest here is a variation of this procedure that selects a subset of any data set such that the maximum error variance of this subset is less than a specified limit. The subset for this procedure is initialized as empty. Sampled points are added to this subset one at a time in a specific order and the maximum error variance of this subset is computed after each point is added. This continues until the limiting maximum error variance is reached, at which time the points added and their specific ordering are saved. Every other possible order of the sampled points is processed similarly. The ordering that reached the limiting variance with the fewest added sampled points is selected as the minimal data set with this specified error variance. Since this is an $O(n^2)$ process, Szidarovszky also advocates a

branch-and-bound implementation of this technique to minimize the search for the minimal data set. This can be done because of the monotonic properties of the error variance: the maximum error variance for a given data set is never less than the maximum error variance for that same data set after adding any other sample (47:334).

A less than optimal method of building this minimal data set developed by Brodkin (2) is to add points to the minimal data set one at a time. The maximum error variance of the initial minimal data set is computed and the next sample is added at the position where the maximum error variance occurs. This process is repeated using the new minimal data set if the limiting maximum error variance has not been reach. This method does not guarantee that the minimal data set contains the minimum number of sampled points necessary to reach this limiting maximum error variance, for it is baced upon the assumption that the point that decreases the maximum error variance the most is the same point that has the highest current error variance (2). Also, the effects of combinations of points are not considered. For example, the addition of point X or point Y separately may decrease the error variance less than the addition of any other point, but the addition of both point X and Y may actually decrease the error variance by a greater amount than any other combination of points. This method may never add the first of these points to the minimal data set in order to discover the effect of the pair.

In order to facilitate either of the above methods of constructing a minimized data set, it is useful to introduce the method of *inversion by blocks* or *inversion by partitioning* (47:336)(48:192-5) in order to solve almost identical systems of equations. Each time additional points are added to the minimal data set, a new value for the error variance $\sigma_E^2 = w^T B$ must be computed; correspondingly, a new $w = A^{-1}B$ must be computed (see Eqs (2.11) and (2.10), respectively). Each of these solutions must invert a new A' that is almost identical to the previous A; A' simply contains extra rows and columns corresponding to the additional points being considered in the minimum data set.

To apply the method of partitioning by blocks, the A matrix must be rearranged such that the new rows and columns can be added to the bottom and right, respectively:

$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & \cdots & 1 \\ 0 & 0 & 0 & X_{s_1} & X_{s_2} & \cdots & X_{s_n} \\ 0 & 0 & 0 & Y_{s_1} & Y_{s_2} & \cdots & Y_{s_n} \\ 1 & X_{s_1} & Y_{s_1} & \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1n} \\ 1 & X_{s_2} & Y_{s_2} & \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & X_{s_n} & Y_{s_n} & \sigma_{n1} & \sigma_{n2} & \cdots & \sigma_{nn} \end{bmatrix} \tag{3.2}$$

If, on a given step, k points are added to the minimal data set, k rows and columns must be added to A to obtain A'. This A' matrix can now be represented in terms of A as:

$$A' = \begin{bmatrix} P & Q \\ R & S \end{bmatrix} \tag{3.3}$$

where $P = A$, Q is a $n \times k$ submatrix containing the appropriate covariances between the n original points and the k added points, $R = Q^T$, and S is a $k \times k$ square submatrix containing the appropriate covariances between each of the k added points only. Using Eq (3.3), A'^{-1} can be obtained with the following equation:

$$A'^{-1} = \begin{bmatrix} Y & Z \\ U & V \end{bmatrix} \tag{3.4}$$

where

$$\begin{aligned} V &= (S - RP^{-1}Q)^{-1} = (S - RA^{-1}Q)^{-1} \\ Z &= -(P^{-1}Q)V = -(A^{-1}Q)V \\ U &= -VRP^{-1} = -VRA^{-1} \\ Y &= P^{-1} - (P^{-1}Q)U = A^{-1} - (A^{-1}Q)U \end{aligned}$$

Since the original A^{-1} is already known, the A'^{-1} can now be computed with Eq (3.4) above by inverting one small matrix and applying matrix arithmetic. This solution is much less complicated and more stable than if A'^{-1} had to be inverted using a method such as LU decomposition and backsubstitution.

A minimized data set derived in either of the above fashions can be thought of as characterizing the entire data set. The accuracy of this characterization can be judged by the error variance resulting from estimating the original data set from this minimized data set; this error variance was the limiting factor in building the minimal data set as described above. As such a characterization, this minimal data set should exhibit characteristics of the terrain under consideration without exactly representing terrain. A grid of estimated points at any resolution can be constructed from this minimized data set using standard kriging techniques. This kriged data set should be a good approximation of original terrain.

If the purpose of constructing a minimal data set is to provide a minimal sample set that can be used to estimate other terrain elevations, certain properties can be taken advantage of to calculate the minimum data set more efficiently. If any point is to be estimated based on this minimal data set, at least one point from the minimal data set must be in the neighborhood of the point being estimated. As this thesis uses the range of the spherical variogram to define the neighborhood, no two points in the minimal data set should be separated by over twice this range. An initial pattern of sampled points can therefore be added to the minimized data set before the above procedure starts, although some optimality may be sacrificed. An 'X' pattern of five known data points, with the length and width of the pattern equal to twice the range, may work well as an initial minimized data set. Additionally, since the error variance is only a property of distances and not of spatial locations, any points that are required within one pattern to minimize the error variance during the minimization process can also be added to the other patterns as well.

3.5 Anisotropy

Anisotropy is present in the sampled data when the data shows different characteristics in different directions or in different zones. Either of these conditions results in

different covariance functions (or semivariance functions or semivariograms) when generated in different directions for the same data set (41). Kriging can be implemented to use the appropriate variogram model depending on the direction between the estimated point and each sampled point, but this would involve many variograms and an inordinate amount of computation. Since it is desirable to obtain one variogram model to characterize the entire data set, anisotropy must be handled.

Chapter II discusses zonal and geometric anisotropy in a regionalized variable. With spherical variograms, geometric anisotropy is exhibited by different ranges in different directional variograms, while zonal anisotropy is exhibited by different sills (41). The method of applying an anisotropic correction factor k as described in Chapter II would account for geometric anisotropy but not for zonal anisotropy as the k-factor only scales the range of the variogram, not its sill. Another method of accounting for geometric anisotropy is to combine the two variograms obtained from orthogonal directions, say, x and y; the x variogram is used on the Δx portion of the distance h between the sampled point and the estimated position and the y variogram is used on the Δy portion of the h.

The arbitrary division of elevation data into blocks along cell or sub-cell boundaries may create blocks with several anisotropic zones. Zonal anisotropy may partially be accounted for by partitioning the block set into homogeneous zones and applying kriging separately to each zone to obtain estimates. One method of partitioning gridded data is described as follows:

1. Sum the values over every row and every column.
2. Determine a median of these sums.
3. Label each row/column as to whether its sum is higher or lower than the median.
4. Partition the data set along horizontal boundaries where the row sums transition from above the median to below the median or from below the median to above the median. Create vertical boundaries in a similar fashion with column sums.

As this may cause very small partitions, a step may be added between steps 1 and 2 above, such that the row/column sum used for a particular row/column is actually an

average of the row/column sums within a window of width n of the particular row/column; n is adjusted to obtain partitions of adequate size. This additional step may not always consolidate smaller zones if they occur in a periodic fashion. Also, this method may not remove all zonal anisotropy; it should partition most data sets into homogeneous zones of a rectangular shape, but the true anisotropic zone may have a diagonal border.

3.6 Summary

This chapter showed how the kriging techniques reviewed in Chapter II could be applied to terrain elevation data in order to estimate terrain elevations, a necessary step toward producing a polygonal terrain model whose vertices may not coincide with the original gridded DTED data. Also shown were methods of minimizing a terrain elevation data set with respect to an inherent measure of accuracy, the error variance. These methods provide a basis for the implementation of tools to model terrain with a specified accuracy (as measured by the error variance) or at a specified level of detail (as measured by the number of polygons); these tools are further discussed in Chapter IV.

IV. Implementation of the Terrain Modeling Systems

This chapter covers the system requirements and design of the terrain modeling software system produced by this effort, along with various implementation issues encountered during its development.

The primary emphasis of this thesis, as presented in the research objectives listed in Chapter I, is to produce terrain models whose polygonal vertices may or may not coincide with elevation posts provided in DTED. Elevations must be estimated for locations that do not coincide with known elevations posts. Therefore, the kriging estimation methodology discussed in Chapter III provides a primary part of this system.

4.1 System Requirements

The system implemented by this thesis uses DMA Level 1 DTED to produce a polygonal model of the terrain represented by that DTED. The following is a list of requirements and constraints for such a terrain modeling system; these requirements are based on the research objectives listed in Chapter I.

- DMA has traditionally distributed DTED on 9 track tape, but most current distributions or made on CD-ROM. The system should be able to read and use DMA Level 1 DTED from either source.
- To support systems using Brunderman's GDMS (3), the final format of the terrain model should be the AFIT Geometry File format, hereafter referred to as a .geom file. The .geom file contains a polygonal description of an object or objects that can be rendered by various tools at AFIT (16).
- Brunderman's GDMS also requires that *database files* be created. These files specify the relationship of the terrain gridsquares and the locations of the actual .geom files corresponding to each resolution of each gridsquare. The user should also be able to set the distances from the viewer that different levels of resolution are to be used by the terrain renderer, as this information is also a part of the database files. These

files were developed by Brunderman (3) for the GDMS and their formats can be found there.

- The user should be able to control the resolution of the model with an intuitive control mechanism. This resolution should correspond to the size and number of polygons in the model. The user should be able to specify *any* resolution, not just those that can be obtained by filtering DTED, in order to finely control the trade-off between rendering speed and image realism.
- The user should be able to specify up to three different model resolutions that can be combined into a multiple-level-of-detail terrain model for use in a rendering system that can handle such models. Allowing three levels of detail for such models is in accordance with requirements of Brunderman's GDMS (3).
- When using multiple-levels-of-detail in conjunction with a terrain model, different parts of the model are usually at different distances from the viewer, and there is no fixed distance to base the determination of which single level of detail to use. Therefore, Brunderman's GDMS requires that the terrain model be divided into *grid squares*, where each grid square is a self-contained model that can be rendered independently of the other grid squares at the level of detail appropriate for its distance from the viewer. The user should be able to specify the size of these grid squares.
- The terrain model should use color to provide the viewer a sense of realism and scale.
- The entire system should be easy to use, with an intuitive interface and understandable controlling parameters.

4.2 System Overview

The resulting terrain modeling system was constructed in two phases: Phase I provided a complete functioning system based on data filtering techniques; Phase II added kriging estimation techniques into this system in order to estimate terrain elevations between DTED posts. This two phased approach ensured that any problems encountered while implementing kriging were indeed kriging problems, and it also provided a means

early in the thesis cycle to produce simple terrain models for use while developing the systems that use Brunderman's GDMS. This was also a natural system evolution, as filtering techniques could only provide a finite set of resolutions from Level 1 DTED, but kriging could provide any resolution desired using estimation techniques.

The basic system was implemented in a piecemeal fashion. Each piece is a standalone `Unix` program, allowing a incremental approach to the implementation of the system. Most of these programs can, unless otherwise instructed, accept input from `stdin` and write output to `stdout`, and almost every program uses the `.pts` file format as a common information interchange format as described below and in Appendix E. This makes it possible to use `Unix` pipes and create a pipeline of programs, avoiding the creation of very large intermediate files. To provide a simplier interface, the total modeling pipeline can be invoked using another program that invokes the other programs in the proper order with the proper parameters.

All software developed for this terrain modeling system is written in `C` and `C++` and, except where noted, was compiled with the gnu `g++` compiler on a Sun 4/260 Workstation. This software is available upon request.

4.3 Phase I: Terrain Modeling using Filtering

4.3.1 Overview. The first phase implemented a set of four programs: `tape2dted`, `dted2pts`, `pts2geom`, and `connect`, as shown in Figure 4.1. This is called the *basic terrain modeling pipeline*. Each of the four programs in this `Unix` pipeline provides some part of the translation from DTED to a polygonal terrain model required for Brunderman's GDMS. Each of these programs is described in detail below; comprehensive guides to their use can be found in Appendix D. This entire pipeline can be controlled using a program called `tb`; this program is also discussed below and in Appendix D.

4.3.2 `tape2dted`. The program `tape2dted` reads DTED from an `Unix` device such as a 9-track tape drive and produces a binary data file of the next DTED file read from that device. `tape2dted` assumes that the tape contains binary DTED files formatted as specified by DMA (14); each file should contain gridded terrain elevation data for a $1^\circ \times 1^\circ$

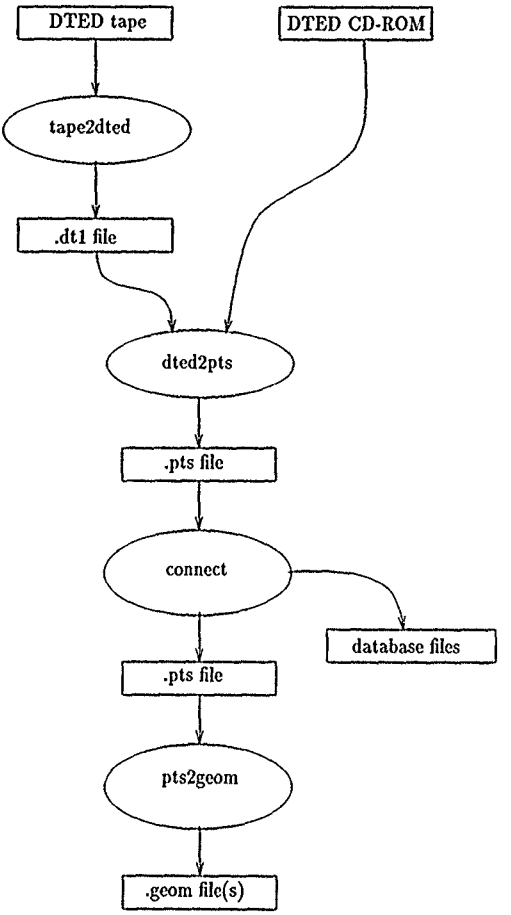


Figure 4.1. Phase I Pipeline

DTED cell as defined in Chapter II. This program reads the next such file encountered from the current tape position and writes the elevation information to the specified output file (or `stdout` if unspecified). Alternately, it can be instructed to read every file from the current tape position to the end of the tape; in this case the output is written to files with pre-determined names based on their latitude and longitude. As this later case creates multiple output files, it breaks the `Unix` pipeline as described above, but it does provide an efficient method of removing all DTED data from a DTED tape.

DMA currently distributes DTED in CD-ROM format as well, but the basic file structure is very similar between the two medias, However, the files are accessible from the CD-ROM through an ordinary `Unix` file structure and can be manipulated via simple `Unix` commands. Therefore, `tape2dted` creates output files in the DTED CD-ROM file format. As DMA uses the `.dt1` filename extension for these files, these files are referred to as `.dt1` files. The use of this `.dt1` file format provides a standard file structure that can be easily created from DTED tapes or simply copied from DTED CD-ROM.

Although shown as part of the `Unix` pipeline of Figure 4.1, `tape2dted` is more easily run separately, especially since `tape2dted` must deal with an I/O device that may encounter occasional problems. Also, available machines with 9-track tape drives may not have the power to perform some of the functions necessary in the other pipeline programs in a reasonable time. As noted above, the `Unix` pipeline approach is not supported if `tape2dted` is instructed to read until end-of-tape and produce a `.dt1` file for each DTED file encountered. For this thesis, `tape2dted` was run separately from the rest of the pipeline and a library of `.dt1` files were created for later use.

`tape2dted` was compiled and run on a MicroVax III; compilation was performed with the gnu `g++` compiler. The program is based on a program originally written by Roberts (40).

4.3.3 `dted2pts`. The `.dt1` files created by `tape2dted` or obtained from DTED CD-ROM are still in a raw binary form. The program `dted2pts` restructures this data into the `.pts` file format for further manipulation by other programs in the terrain modeling pipeline (see Appendix E). The other programs in this system could have been written

to accept any file format, and a binary format would have been quicker in several cases. However, the programs that performed kriging estimation were written by a group of AFIT thesis students to process data from several different sources; a basic ASCII file structure providing data in `x y z` format was agreed upon as the common format for these programs.

`dted2pts` can accept input from a specified `.dt1` file or from `stdin` if no input file is specified by the user; correspondingly, output is written to the specified `.pts` file or to `stdout` if no output file is specified by the user.

The `dted2pts` program also allows the user to select only a portion of the data in the `.dt1` file to be written to the output `.pts` file by specifying limiting latitudes and longitudes; the elevation data within these latitudes and longitudes must be entirely contained within the specified DTED file. Simple filtering of the data can also be performed by `dted2pts`; the user can specify the desired spacing (or resolution) in arc minutes between elevation posts. As this program performs no interpolation between elevation posts, elevation posts must exist in the `.dt1` file at the spacing specified; if there is no such correspondence, the spacing is reduced until it does correspond with DTED elevation posts. For convenience, `dted2pts` considers all DTED elevation posts as separated by 0.05 arc minutes; DTED that does not maintain this post spacing (such as that above 40° north latitude and below 40° south latitude), would be interpreted by `dted2pts` as if it did.

4.3.4 `pts2geom`. The program `pts2geom` processes the elevation data contained in a `.pts` file and creates one or more `.geom` files containing a polygonal terrain model based on this data. Although the `.pts` file format allows for gridded or ungridded data, `pts2geom` requires that the data in the input `.pts` file be gridded and arranged in column-major form; all other programs in this system writes `.pts` file data in this format. `pts2geom` can build one `.geom` file from a `.pts` file, but it can also sub-divide a terrain description contained in a single `.pts` file into *grid squares*, each in a separate `.geom` file, as described in the requirements above. These grid squares can be linked back together by `connect` to produce a large terrain model (see Section 4.3.5 below). This technique is faster than dividing the a cell of terrain data into blocks using `dted2pts` and processing each block through `pts2geom`.

If an input .pts file is not specified, pts2geom reads input data from stdin. However, output is written to stdout only if no grid square division is desired; otherwise multiple output .geom files are created, one for each grid square, named in accordance to a user specified format. Since pts2geom is designed to be run as the last stage in the Unix pipeline described above, not writing to stdout does not break this pipeline.

As stated earlier, the .pts file that is used by pts2geom must contain gridded terrain elevation data in column-major order as specified in Appendix E. This gridded data is placed into a C++ object called Grid. Triangular polygons are created from the Grid object by connecting the nodes within this Grid as described in Chapter II such that the grid nodes become polygon vertices. The .geom file requires that a normal vector be specified for each vertex; these normals are determined by averaging the normals calculated for the adjacent polygons that use this node as a vertex. There is a possibility that seams will be apparent in the terrain model along the edges of contiguous blocks that were built separately; this is caused when polygons on either side of this edge share vertices, but the normals at these vertices were calculated based on a different set of adjacent polygons. These seams are usually minor, but they can be avoided for the most part if the entire DTED cell is sub-divided by pts2geom into multiple .geom files rather than processing it through the entire Unix pipeline in pieces; pts2geom computes all normals blocks before any divisions are made, eliminating the seams.

The polygonal vertices extracted from the grid class, along with normals, connectivity information, and colors, are placed in the destination .geom file. The x, y, and z values of these vertices may be scaled if desired. Some scaling may be necessary on .pts files generated from DTED data since DTED encodes x and y values in arc minutes (nautical miles) and z values in meters. In such cases a scaling of approximately 1860 on x and y produces true-scale terrain. However, true-scale terrain models often do not appear realistic (20:45)(31); therefore, the terrain surface elevation (z-value) may also need to be exaggerated somewhat to produce visually pleasing images from the terrain models.

The basic color of any resulting polygons is a factor of the z-value of its highest vertex after scaling; alternate triangular polygons are colored slightly different to create an illusion of texture. As described in Appendix D, colors are specified in red/green/blue

tuples (referred to here as *rgb tuples*) with each component ranging between 0 and 1. Polygons are colored white (rgb=(0.8, 0.8, 0.8) or rgb=(0.7, 0.7, 0.7)) if their highest z-value is at or above 3000 units; polygons with z-values between 0 and 3000 units are colored dull green (rgb=(0.62, 0.55, 0.0) or rgb=(0.7, 0.65, 0.0)); polygons with a z-value at 0 units are colored blue (rgb=(0.2, 0.2, 0.8) or rgb=(0.2, 0.2, 0.7)); polygons with z-values below 0 units are colored brown (rgb=(0.7, 0.4, 0.2) or rgb=(0.7, 0,3, 0,2)).

As an option, polygons in the .geom file may be specified with a color at each vertex; the rendering system should blend the colors specified at each vertex across the polygon surface. This allows the option of using a range of colors associated with the terrain elevation that varies continuously. If this option is specified, the vertices are colored according to the rgb tuple calculated by the following heuristicly developed formula:

$$\text{red} = \begin{cases} 0.9 + r1 & \text{if } z \geq 3000 \\ 0.9\left(\frac{z}{3000}\right) + r1 & \text{if } 0 < z < 3000 \\ 0.2 + r1 & \text{if } z = 0 \\ 0.7 + r1 & \text{if } z < 0 \end{cases}$$

$$\text{green} = \begin{cases} 0.9 + r2 & \text{if } z \geq 3000 \\ 0.3 + 0.6\left(\frac{z}{3000}\right)^2 + r2 & \text{if } 0 < z < 3000 \\ 0.2 + r2 & \text{if } z = 0 \\ 0.4 + r2 & \text{if } z < 0 \end{cases}$$

$$\text{blue} = \begin{cases} 0.9 + r3 & \text{if } z \geq 3000 \\ 0.9\left(\frac{z}{3000}\right)^3 + r3 & \text{if } 0 < z < 3000 \\ 0.7 + r3 & \text{if } z = 0 \\ 0.1 + r3 & \text{if } z < 0 \end{cases}$$

where z is the elevation at the particular vertex and $r1$, $r2$, $r3$, and $r4$ are random numbers between 0.0 and 0.1. Before the random numbers for a particular vertex is generated, the random number generator is always seeded using a function of the x and y of that vertex; this allows vertices that may be repeated on several polygons to always be colored the same. This coloring scheme provides the appearance of a smooth transition from white at

an elevation of 3000 units through a dull green to a deep green at 0 units; vertices at 0 units are blue and vertices below 0 units are brown. The random components are added to each color to provide some texture.

4.3.5 `connect`. Brunderman's GDMS requires two additional files in order to use the polygonal terrain model generated by the above programs. These files, referred to as `.dbs` and `.lnk` files, are collectively known as the *database files*. The database files provide various connectivity and level of detail information. The primary purpose of these files is to allow multiple polygonal descriptions in different `.geom` files to be used together, either as different terrain grids in a large area or as different levels of resolution of the same grid. These database files are fully described in Brunderman (3).

`connect` builds the database files in an incremental fashion. Provided with the name of a `.pts` file and the `.geom` files that were created from it by `pts2geom`, this program adds the necessary information from these files into the database files; the database files are created if they do not already exist. The input `.pts` file is read from `stdin` if not specified. The base name of the database files must always be specified; these filenames are differentiated only by their `.lnk` and `.dbs` extensions.

As the specified `.geom` file names are simply added to the database files without checking for their presence, they do not need to exist until the database files are used to render them. Also, `connect` writes the input `.pts` file to `stdout` without modification if (and only if) it was read from `stdin`. This allows `connect` to be run before `pts2geom` in the `Unix` pipeline; this is important since `pts2geom` outputs `.geom` files instead of `.pts` files.

This program also allows x, y, and z scaling factors to be placed into the database files associated with each terrain grid added. Other optional information placed in these files by these programs include a flag indicating the level of resolution particular the `.geom` files represents (either low, medium, or high) and the distances from the viewer where transitions should occur between the level of details provided.

4.3.6 `tb`. The programs listed above can be used as stand-alone utilities that each read their respective input files and produce output files that other programs in the system can use; alternately they can be used in pipeline fashion as described above. However, when used as stand-alone utilities, they create files that take up enormous amounts of disk space, and when used as a `Unix` pipeline, the user interface can be a bit cumbersome. The `tb` program was written as a shell around this system in order to provide a simple user interface. `tb` does not actually invoke the terrain modeling programs in as a `Unix` pipeline; rather, the programs are invoked individually and temporary files are removed when they are no longer needed. This program resembles an `Unix` shell script, but it is actually written in `C++`. `tb` reads some parameters from the command line, but several of the more general parameters are read from a configuration file. The user provides the desired latitude, longitude, resolution and output filename for the resulting terrain model. Detailed information on its use is provided in Appendix D.

4.4 Phase II: Terrain Modeling using Kriging

4.4.1 Overview. The second phase of this thesis implemented five additional programs: `resid`, `varfit`, `krige`, `rebuild`, and `partition`. These program are designed to be incorporated into the `Unix` pipeline described in Phase I above. Each of these programs perform some part of the kriging methodology as described in Chapters II and III. They are each described in detail below, and comprehensive guides to their use can be found in Appendix D. The revised modeling pipeline is shown in Figure 4.2.

Four of these programs (`resid`, `varfit`, `krige`, and `rebuild`) were designed to accept non-gridded (randomly spaced) or gridded data in support the other projects that used kriging during this thesis cycle at AFIT(2)(30). The fifth program, `partition`, depends on the data being gridded to successfully partition it. Both `krige` and `varfit` take advantage of the regularity of the data if it is gridded, but they can perform their functions on non-gridded data as well. As each of the three thesis efforts referenced above use different data formats and had different reasons to use kriging, each of the programs listed in this section were built to use different interface functions to read and write the data in the format they required. In this effort, all five programs read data from and write data to a `.pts`

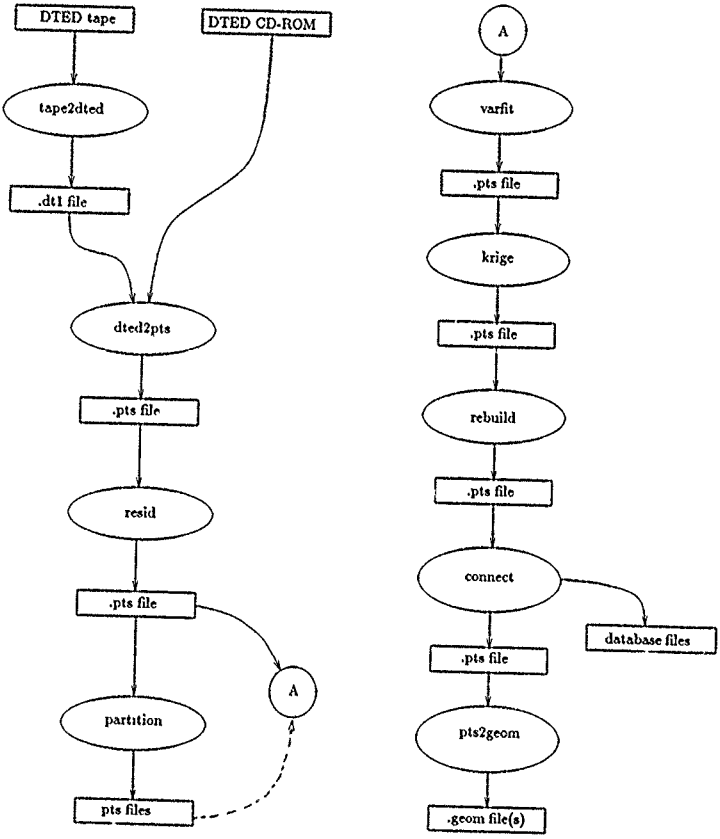


Figure 4.2. Phase II Pipeline

file; creating a consistent interface with the programs in the original modeling pipeline presented above. Control information is read from the input `.pts` file header; most control information read and all control information generated by these programs is written to the output `.pts` file header.

The programs `resid`, `rebuild`, and `partition` were written in conjunction with McGee (30) and Brodkin (2) who used them for similar thesis efforts. The programs `varfit` and `krige` are modified versions of programs used by Grant (19) as provided to this thesis effort by Robinson (41); modifications were performed in conjunction with McGee (30) and Brodkin (2). As most modifications to `krige` were performed by Brodkin (2), the actual implementation details can be found there.

4.4.2 `resid`. The `resid` program insures that the terrain data used to produce kriged estimates is globally stationary. This program reads terrain elevation data from a `.pts` file and extracts global trend by fitting a two-dimensional second-order polynomial to the data using least-squares regression. The second order polynomial used, as adapted from Eq (2.14), is of the form:

$$M_{\texttt{global}} = a_0 + a_1 x + a_2 y + a_3 xy + a_4 x^2 + a_5 y^2 \tag{4.1}$$

This trend polynomial is subtracted from the elevation data before this data is written back out to the output `.pts` file. The coefficients a_i of the trend polynomial fitted to the data are written to the header of the output `.pts` file; `rebuild` later uses this information to add the trend polynomial back to the kriged elevation data (see Section 4.4.5 below). In support of the `Unix` pipeline, input is read from `stdin` if no input `.pts` file is specified, and output is written to `stdout` if no output `.pts` file is specified.

Because of the computational cost of fitting a second-order polynomial to a large data set, a pre-determined trend polynomial may be extracted from the data instead. In this case a second `.pts` file should be specified. This second file should already contain the polynomial trend coefficients in its header information. It is this polynomial that is extracted from the data contained in the first `.pts` file; no polynomial fitting to the input data is actually performed. The coefficients in the header of the second file could be placed

there by hand, but the expectation is that this second data set is a filtered version of the first (created by specifying a larger gridsize when running `dted2pts`) that has already been processed by `resid`.

It is not necessary to remove global trend before kriging, although the resulting estimates may not be optimal. Therefore, this pipeline step may be skipped if speed is of a greater concern than accuracy.

4.4.3 `varfit`. The `varfit` program generates directional experimental variograms and variogram model parameters based on data read from the specified input `.pts` file; if no input file is specified, the input is read from `stdin`. Two experimental variograms are generated as described in Chapters II and III for both the northerly (0°) and easterly (90°) directions unless otherwise specified. If a variogram at a different direction is desired, only the single specific directional variogram specified is generated.

For this thesis, most `.pts` files used to generate variograms contain gridded data, but `varfit` can also generate variograms for irregularly spaced data as well. If gridded data is used and the variogram is being generated at 0° or 90°, the variogram is generated using a more efficient method of selecting pairs of points from rows or columns of the grid. Otherwise, the data is exhaustively searched several times to generate any one variogram. Both methods are described in Chapter III.

`varfit` fits three variogram models to the elevation data; the linear, De Wijsian, and spherical variogram models are all fitted to the resulting experimental variograms using a least-squares regression method. The coefficients of these models are written to the header of the output `.pts` file. The angle that each particular variogram model is generated from and a simple correlation signifying how well each particular model fitted the experimental vario· m are also written to the `.pts` file header. This allows several variogram models based on several different directions to be retained in one `.pts` file. The remaining elevation data from the input `.pts` file is simply copied to the output `.pts` file for later use by `krige` (see Section 4.4.4 below); if no output `.pts` file is specified, output is written to `stdout`. It should be noted that, even though three different models are fitted to the data, `krige` only uses the spherical models generated at 0° and 90° to perform the

desired estimations.

As mentioned in Chapter III, the variogram model fitted to the experimental variogram may not be adequate for kriging. Preliminary testing showed that the spherical variogram model seemed to exhibit more error near $h = 0$ when compared to the experimental variogram to which it was fitted, artificially raising or lowering the nugget effect. If the size of the neighborhood used during kriging is small with respect to the extent of the experimental variogram, most weight calculations will be based on this erroneous part of the variogram. To help catch cases like this, `varfit` checks for a negative nugget C_0; it is reset to zero if it is only slightly negative with respect to the variogram sill, or the program exits with an error message if the nugget is negative with a magnitude greater than 10% of the sill. Errors are also produced if the range a is negative or the nugget C_0 is greater then the sill $C + C_0$. Other inaccuracies in the variogram model may cause problems during kriging, but these must be caught by inspection and the variogram model must be modified by hand.

In addition to the output `.pts` file, `varfit` optionally produces a descriptive *variogram file* containing a listing of the experimental variograms and the variogram models generated, along with other information from the variogram fitting process. A *plot file* is also produced if desired; this file simply contains the x, y pairs of the experimental variogram.

4.4.4 **`krige`.** The program `krige` accepts as input a standard `.pts` file that contains spherical variogram model parameters for both 0° and 90° as generated by `varfit`. If this file is not specified, input data is read from `stdin`. This program performs two basic functions: the kriging estimation of elevations at positions not provided in the input `.pts` file, or the selection of a subset of elevation posts from those provided in the input `.pts` file such that the error variance is minimized.

- **Kriging Estimation.** The basic function of `krige` is to estimate the elevations at unsampled points directly using the methods described in Chapters II and III. Estimates are made based on the sampled data and spherical variogram model parameters provided through the input `.pts` file. This input data need not be gridded,

but the estimates are always made at the vertices of a regular grid whose spacing is specified by the user. If vertices from this new grid coincide with points in the input `.pts` file, the elevation value at these points are used in place of the estimates; otherwise the weighted sum estimator as described in Chapter II is used. The error variances of the estimates can also be written to a separate `.pts` file for additional analysis if desired.

- **Error Variance Minimization.** Several methods for data set minimization based on the error variance or largest difference were presented in Chapter III; this program implements the two methods developed by Brodkin (2). The first method creates a minimal data set based on adding points from the original data set at positions of maximum error variance until the maximum error variance is less than a given threshold called the *maximum variance.* The technique presented in Chapter III of replicating a pattern of initial points over the output grid and adding new points simultaneously to each of these patterns is used by this program.

  The second method minimizes the data set by adding points from the original data set at positions where the difference between the actual elevation and the kriged estimate of the elevation is maximum; this is continued until this maximum difference is less than a given threshold called the *largest difference.* This method also uses the pattern technique described in Chapter III. Both methods are described in Brodkin (2).

This program uses universal kriging with linear drift in both of the above applications to estimate elevations at unknown positions and to compute an error variance associated with the estimations; however, the program is easily modified to account for either quadratic drift or no drift. The neighborhood used during estimation is defined by the range of the spherical variogram model as described in Chapter III; this range may be artificially constrained to obtain quicker but less accurate estimates. The sill of the spherical variogram model is used as the variance of the data set; this value is used to compute the covariances needed in the kriging system of equations as presented in Chapter II. The linear and De Wijsian variogram models, although provided in the `.pts` file by `varfit`, are not used by this program.

As described in Chapter III, the estimation process implemented in `krige` may be computationally intractable when applied to the reduction of dense terrain elevation data. Therefore, the neighborhood size used by `krige` to estimate an elevation can be artificially constrained to obtain a less then optimal estimate where an optimal estimate would be either too time and CPU intensive. One method of constraining the neighborhood is to lower the actual range of the spherical variogram model used to perform estimation. This would affect the optimality of the resulting estimations not only by restricting the number of points in a neighborhood but also by changing the covariance function that the remaining kriging weights are based on. Another way `krige` allows the neighborhood to be constrained that does not change the covariance function is by limiting the buckets searched when determining the points in a neighborhood. All elevation data in `krige` is stored in buckets corresponding to output grid vertices (see Section 3.3) and these buckets are searched for points that may lie in a estimated point's neighborhood by starting with the bucket corresponding with that estimated point and working outward. `krige` allows the user to specify how many steps outward this bucket search may progress, but as a safeguard it automatically limits the number of steps to include not more than 250 points. This is a course constraining control, as a single bucket may still contain too many points to perform graceful estimation.

As described in Chapter III, geometric anisotropy is handled in `krige` by using an anisotropy ratio k. The x component of the distance between the position with known value and the position to be estimated is scaled by $k = \frac{y\text{-}range}{x\text{-}range}$ before using the distance to compute a semi-variance or covariance. However, this program is not robust enough to handle zonal anisotropy effectively. As zonal anisotropy is characterized by different sills in different directional variograms, `krige` simply averages the two sills to avoid this problem. It is recognized that this may make the estimates slightly less optimal, but it is the only effective way to automate the disposition of zonal anisotropy

4.4.5 `rebuild`. Any elevations estimated by `krige` are based on the residuals of the sampled elevation data as calculated by `resid`. Therefore, the global trend must be added back to the kriged `.pts` file in order to have a true representation of the terrain.

The program `rebuild` reads the polynomial trend coefficients from the header of the input `.pts` file and adds the trend polynomial back to the elevation data in the file. Input is read from `stdin` if no input `.pts` file is specified; the results are written to the output `.pts` or to `stdout` if no output file is specified. The polynomial trend coefficients are reset to zero in the output `.pts` file so that an inadvertent second pass will not corrupt the elevation data.

As stated above, it is not necessary to remove global trend before kriging. If `resid` is not run on the data before kriging, no polynomial trend coefficients are placed in the header of the `.pts` file, and `rebuild`, if run, will not add anything to the resulting elevations. Therefore `rebuild` can be considered an optional pipeline step if `resid` is omitted.

4.4.6 `partition`. In order to attempt to handle zonal anisotropy, `partition` partitions the terrain elevation data from an input `.pts` file into multiple output `.pts` files according to the following method originally presented in Chapter III:

1. Sum the values over every row and every column.
2. Reset the values of the rows/columns sums to the average of the rows/columns sums within a window of width n around that particular row/column; n is provided by the user.
3. Determine the median of the row sums and the column sums.
4. Label each row/column as to whether its sum is higher or lower than the median.
5. Partition the data set along horizontal boundaries where the row sums transition from above the median to below the median or from below the median to above the median. Create vertical boundaries in a similar fashion with column sums.

This method divides the block of terrain elevation data in a `.pts` file into multiple sub-blocks that are homogeneous with respect to their average elevation. The input can be accepted from `stdin`, but each of the sub-blocks are written to a separate output `.pts` file; the files are named according to a user-specified filename format. As output cannot be written to `stdout`, this program, if used, breaks the terrain modeling pipeline; multiple

smaller pipelines may be re-started using each of the resulting `.pts` files. This pipeline break is shown as a dashed line in Figure 4.2.

4.5 Summary

This chapter presented an overview of the tools and utilities implemented as parts of the polygonal terrain modeling system referenced in the research objectives in Chapter I. Implemented in two phases, all the tools described above are designed to interface with each other via the `.pts` file and its header. Used together, these tools build a polygonal terrain model based either on simple filtering of the DTED or on kriging estimations of elevations not provided by DTED. These tools are used in Chapter V to demonstrate the effects of both forms of terrain modeling.

V. Results

This chapter presents the results of using the terrain modeling pipelines presented in Chapter IV to generate accurate and realistic polygonal terrain models from dense terrain elevation data. This chapter is organized into three parts. First, the results of using the basic terrain modeling pipeline that implements data filtering techniques are presented in Section 5.1. This system provides discrete control over the complexity of the resulting models, but no direct control over realism is available and there is no measure of accuracy. Second, the results of using the enhanced terrain modeling pipeline that incorporates kriging estimation techniques are presented in Section 5.2. This system allows the complexity of the model to be controlled and ensures the accuracy of the resulting model by guaranteeing that the error variance is as small as possible for a linear weighted sum estimator. Finally, Section 5.3 briefly discusses the use of data set minimization as implemented in the enhanced terrain modeling pipeline to provide a means of controlling the accuracy (as opposed to the complexity) of the resulting model. All of these sections provide images generated from models built with their respective techniques and rendered by Brunderman's Database Generation System (DBGen) (3). All terrain models are based on the four DTED Test Cells as presented in Chapter I and again in Table 5.1 (Note that this is not the same set of DTED Evaluation Cells used in Chapter III to evaluate and support the development of the kriging methodology). This chapter concludes with a summary of the results. These results show that kriging does provide a finer control of a model's complexity than filtering techniques but a degree of realism is sacrificed.

DTED Test Cell #1	36° N x 113° W (Grand Canyon)
DTED Test Cell #2	36° N x 118° W (Death Valley)
DTED Test Cell #3	38° N x 121° W (Sierra Nevada)
DTED Test Cell #4	37° N x 123° W (San Francisco Bay)

Table 5.1. DTED Test Cells

5.1 *Terrain Modeling using Filtering Techniques*

The basic terrain modeling pipeline (the Phase I system shown in Figure 4.1) successfully creates accurate and realistic terrain models for use by systems using Brunderman's GDMS. Figures 5.4 - 5.7 show images of terrain models built using this modeling pipeline. Based on the filtered DTED resolution used by others (23)(31) and on the early performance of the Database Generation System (`DBGen`) (3), a standard terrain model resolution of approximately one elevation post per mile was established for the purpose of testing these terrain modeling systems. Therefore, the models presented here are generated from DTED Test Cells filtered to one elevation post per arc minute of latitude or longitude (approximately one nautical mile); this corresponds to using every 20th data point in the original DTED file. Each model is of an entire DTED cell modeled at a single level of detail. The polygons are colored using the vertex coloring scheme described in Chapters III and IV, and to enhance the realism of the images, the z elevations are exaggerated by scaling them to five times their original values after the x and y values were scaled to the same units as z.

Figures 5.8 and 5.9 are images of terrain models generated from portions of DTED Test Cells #1 and #2, respectively, using elevation posts at a higher level of resolution than above. Each model is built from a 10 arc minute (latitude) by 10 arc minute (longitude) area from their respective DTED cell as selected by `dted2pts`; no filtering of the elevation data was performed. The first model covers an area from 36°20′ N to 36°30′ N latitude and from 112°30′ W to 113°40′ W longitude as extracted from DTED Test Cell #1, while the second model covers an area from 36°20′ N to 36°30′ N latitude and from 117°30′ W to 117°40′ W longitude as extracted from DTED Test Cell #2. For comparison, Figures 5.10 and 5.11 are images from models built from the same terrain areas filtered to every 20th elevation post. As before, all of these models are at a single level of detail and use vertex colors; elevations are also exaggerated to five times their original values. These images demonstrate that terrain models of varying sizes and resolutions can be generated using this modeling pipeline.

Figure 5.12 shows the use of a multiple-level-of-detail terrain model built by the basic terrain modeling pipeline and rendered by `DBGen`; this image demonstrates the ability of

this terrain modeling system to build an accurate and realistic terrain model while controlling the model's complexity through the use of multiple-level-of-detail. This image also demonstrates the coloring of polygons in an alternating fashion discussed in Chapter IV in order to enhance the terrain texture. As described in Chapter IV, this is done by *not* specifying the use of vertex colors when running `pts2geom`. This image is built from the DTED Test Cell #1. Again, all elevations are exaggerated to five times their original values to obtain this model.

This complete multiple-level-of-detail terrain model was built by first extracting elevation data from the DTED at three different resolutions into three `.geom` files using `dted2pts`. The low level model uses elevation posts separated by 6 arc minutes of latitude or longitude (approximately 6 nautical mile), the medium level model uses elevation posts separated by 3 arc minutes of latitude or longitude (approximately 5 nautical miles), and the high level model uses elevation posts separated by 45 arc seconds of latitude or longitude (approximately 0.75 nautical miles). To be used as a multiple-level-of-detail terrain model, each of these sets of elevation posts were subdivided by `pts2geom` into a 10 × 10 grid of 100 smaller terrain models, each covering a square region of 6 arc minutes latitude by 6 arc minutes longitude; each of these smaller models were placed in separate `.geom` files. `connect` was used to build the corresponding pair of database files in order to mesh the models together for Brunderman's GDMS.

As shown above, the basic terrain modeling pipeline yields models whose complexity is controllable only by varying the resolution that the original terrain elevation data is sampled. However, this sampling resolution can only be modified in discrete steps. Also, the realism of such models seems related to the inverse of this resolution and therefore varies from model to model. No measure of accuracy is available with this modeling system.

5.2 Terrain Modeling using Kriging

This section presents the results of using the enhanced terrain modeling pipeline (the Phase II system shown in Figure 4.2) to generate terrain models. This pipeline uses `resid`, `varfit`, `krige`, `rebuild`, and `partition` to perform kriging estimation in conjunction with terrain modeling. The goal of this system is to model terrain using a

regular grid of triangular polygons where the grid resolution may not coincide with the elevation posts in the original DTED; kriging estimation is performed if a desired grid point does not coincide with a DTED elevation post. This model should allow the complexity and realism to be controlled while ensuring maximum accuracy in terms of minimum error variance. To address problems with this system as they arose, three different variations of terrain modeling are described. First, this system is used to attempt to generate gridded terrain models from unfiltered full-cell DTED. As this proved memory and time extensive, this system is next used to build terrain models from filtered full-cell DTED. Finally, to show what this system could do if time and memory restrictions were not a problem, terrain models are built by kriging only small portions of unfiltered DTED cells. The following sections describe the effects and results of each of these variations.

5.2.1 Kriging From Full Resolution DTED. Kriging can be used to to estimate the elevation of any point that may be necessary in a desired terrain model but that may not be present in the original DTED data. The most basic way of doing this is to build a terrain model covering an entire $1° \times 1°$ DTED cell and to base any kriging estimation on the 1,442,401 (1201×1201) elevation posts of an entire unfiltered DTED cell. An example of this would be to desire a terrain model based on a grid of 62×62 elevation posts (grid spacing of 0.98 arc minutes latitude or longitude) from the full DTED cell. Assuming the desired grid is regularly spaced (an assumption always made throughout this thesis), then only the corner positions of the new grid would have corresponding elevation posts in the DTED cell. Other grid locations would have to be estimated using kriging.

A terrain model as described above would be built as follows. First, the entire DTED cell at its finest resolution would be selected using `dted2pts`; global trend would then be removed from the elevation data using `resid` and a variogram model would be generated based on the resulting residuals using `varfit`. `krige` would then be used to estimate the elevations over the new grid based on the elevation data and the variogram parameters in the `.pts` file. The resulting `.pts` file would contain elevation data based on the new grid, and `rebuild`, `connect` and `pts2geom` would then be run on the elevation data to finish generating the terrain model.

As suspected in Chapter III, a major problem was encountered using this approach: the quantity of data that must be processed when using an entire cell of full resolution DTED (1,442,401 elevation posts) is much too great to be processed by most programs in this system. `dted2pts` does not have a problem with the quantity of data to be processed as it simply filters the elevation data and does not store it; likewise, `rebuild` and `connect` are also designed to handle the dense data. Although `resid` encounters memory problems when using such large amounts of data (as all elevation data must be read in before a global trend polynomial can be determined), this program can be instructed to use a polynomial fitted to a filtered version of the same DTED to remove the global trend from the unfiltered DTED. In this case `resid` needs only write the residual data out as the elevation data is read in and memory problems are avoided. However, the current implementations of `varfit`, `pts2geom` and `krige` cannot avoid memory problems using such large data sets on any machine that was accessible. For example, on a Sun 4/260 with 32 Mbytes of real memory and 50 Mbytes of virtual memory, the process aborted because of memory problems, and on a Silicon Graphics 4D/310 with 8 Mbytes real memory and 40 Mbytes of virtual memory, the process thrashed for over 36 hours and had not even started estimating the first point on the new grid. Modifying the methods incorporated in this system as discussed in the recommendations below may help alleviate these problems.

5.2.2 Kriging From Filtered DTED. Seeing that the original DTED is too dense to be processed by the kriging programs developed for this thesis, a less-optimal method of using kriging to develop a terrain model is to first filter the DTED and use this less-dense data in the kriging processes. An example of this application would be to take a $1° \times 1°$ DTED cell with elevation posts originally spaced by .05 arc minutes of latitude or longitude and extract every 20th elevation post to a `.pts` file with `dted2pts`; this would create a grid of 61 by 61 elevation posts, each separated by 1 arc minute of latitude or longitude. Using kriging to generate a terrain model based upon this new grid is similar to the generation of a terrain model from full resolution DTED using kriging, although memory requirements would be less.

Using the enhanced terrain modeling pipeline to model terrain in this manner brought

to light other problems related to the specific implementation of `krige` and to kriging in general. The range of a spherical variogram model generated from a full resolution DTED cell typically ranges from 10 to 30 arc minutes of latitude or longitude (approximately 10 to 30 nautical miles). (The range and the other spherical variogram model parameters usually vary little between different resolutions of elevation data from a particular DTED cell, but these parameters do vary between individual cells and between a cell and its partitions; the experimental variograms and spherical variogram models fitted by `varfit` for the DTED Test Cells are provided in Appendix C.) `krige` attempts to use this range as the radius of the kriging neighborhoods within this DTED cell. The number of elevation posts within 10 to 30 nautical miles of a position being estimated in full resolution DTED may be over 100,000; even if the DTED is filtered to 1/400th of its original density (as would be the case if every 20th elevation post was used), there may still be 400 elevation posts in the neighborhood of a position being estimated. `krige` must solve a system of equations that contains one unknown for each of the known elevation posts in the neighborhood; a system of equations with even 400 unknowns would take an enormous amount of time to solve even if the LU decomposition and backsubstitution method for matrix inversion could handle it.

For this reason `krige` allows the user to limit the neighborhood size in the manner discussed in Chapter IV and Appendix D. This is also the reason that `krige` limits the number of points considered in any neighborhood to 250. However, limitations such as these cause the resulting estimates to be based upon what can be a inaccurate part of the variogram model near $h = 0$, as shown in Appendix C. McGee (30) actually shows the effects of using a more accurate variogram model than produced by `varfit`; however, this thesis effort relied solely on the variogram models provided by `varfit`. Even allowing for the less optimal estimates resulting from these neighborhood limitations, kriging an entire terrain model in this fashion may still yield a very time intensive problem. However, it does ensure that at least a non-optimal kriged estimate can be determined for every desired point so that a terrain model of some sort can be built and evaluated.

This application of kriging was performed using the four DTED Test Cells listed in Table 5.1 first filtered to a grid of 61 by 61 (a grid spacing of 1.0 arc minutes of latitude

or longitude). Figures 5.13 - 5.16 are images of a terrain models generated from a 62 by 62 grid of elevation posts (grid spacing equal to 0.98 arc minutes of latitude or longitude) that were kriged from DTED Test Cells #1, #2, #3, and #4 respectively, first filtered as described above. Since only the positions at the corners of the new grid corresponded with elevations posts in the filtered DTED, every elevation post in the resulting terrain model except for these corner posts was estimated using kriging. The kriging neighborhood size for all of these models was limited to approximately one-sixth their original size as discussed above. For comparison purposes, the four images generated from these kriged models are based on the same approximate viewpoints and viewing directions as the images generated from the filtered models of the same DTED cells (Figures 5.4 - 5.7, respectively).

Although built from the same DTED, the terrain built with kriged estimates appears smoother than the terrain built by filtering. This is referred to as the "smoothing effect" by David (11). David indicates that this smoothing is unavoidable, since kriging underestimates high values and overestimates low values, resulting in the estimated values being less variable than the real values (11:256). For terrain modeling, this smoothing is undesirable, as many other terrain modeling techniques not only seek to retain the existing detail in a terrain description but seek to enhance or "amplify" it as well (44)(26)(37)(32).

Although this "smoothing" could be used to describe all the differences between the models built by filtering DTED and those built using kriging, other factors inherent in this implementation of kriging may be enhancing this effect. One possibly is that since the kriging estimation is performed using data that contains less of the original terrain data than full resolution DTED, the remaining detail might seem insignificant to the kriging weighted sum estimator. Another possible factor is the limitation that `krige` places on the number of points in a neighborhood of a point being estimated.

One major source that may contribute to this smoothing is poor removal of global trend. The residual terrain elevation data provided by `resid` may not be similar enough to a true regionalized variable, as this data may not be totally stationary. Elevation shaded 2D images of the four DTED tests cells both before and after global trend removal with `resid` are shown in Figures 5.17 - 5.20. Since the removal of trend would result in a more uniform grey level in the image, these figures seem to indicate that a great deal of the

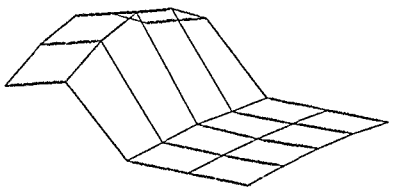


Figure 5.1. The Effect of Trend in Kriging – Original Grid

original trend is still present. An example of how trend may produce a smoothing effect is shown in Figures 5.1 - 5.3. Figure 5.1 is of 25 elevation posts arranged on a 5 × 5 grid such that strong trend is exhibited. Figures 5.2 and 5.3 are of terrain models built from this original gird with `krige` and the other kriging utilities; the kriged grid in Figure 5.2, at 21 × 21, has no vertices coinciding with the original grid vertices, while the kriged grid in Figure 5.3, at 20 × 20, has vertices that coincide periodically. The same smoothing that is exhibited in the terrain models above are apparent in these grids produced by kriging.

Another major source of this smoothing may be that insufficient structural analysis is performed by `varfit` while generating a variogram to characterize the terrain elevation data in a cell. Inspection of various experimental variograms and the models fitted to them by `varfit` (see Appendix C) reveals many inaccuracies in the fitting methodology. Zonal anisotropy may also result from insufficient structural analysis, as the variogram generated to represent the entire surface area may not characterize the terrain equally well in all areas of the cell. The only method implemented of handling any zonal anisotropy that may be present in the elevation data is by using `partition`, and this utility was never fully tested in this effort.

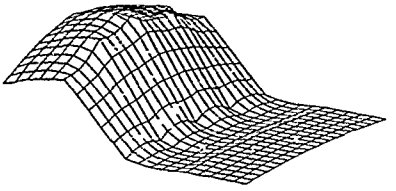


Figure 5.2. The Effect of Trend in Kriging – Kriged Grid at 21 x 21

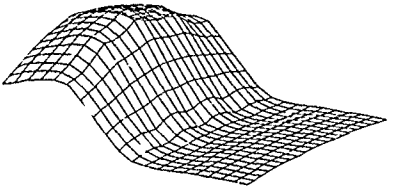


Figure 5.3. The Effect of Trend in Kriging – Kriged Grid at 20 x 20

DTED Test Cell Number	Error Variance	99% Confidence Interval
DTED Test Cell #1	26590.63 m^2	±378.40 m
DTED Test Cell #2	43131.96 m^2	±1246.09 m
DTED Test Cell #3	22308.78 m^2	±896.17 m
DTED Test Cell #4	4447.49 m^2	±400.14 m

Table 5.2. Error Variances for Kriged Terrain Models

Chapter III presented the error variance resulting from the estimations as a possible measure of accuracy for the terrain model generated. The error variances of the individual estimates can be used to calculate confidence intervals for each of the individual estimates as shown in Chapter III. However, it is difficult to make any quantitative statements concerning accuracy based on such a large amount of error information. Similar calculations can be made concerning the entire estimated terrain surface using the maximum error variance for that surface, and Table 5.2 provides the 99% confidence intervals for the four kriged terrain models generated above. The significance of these confidence intervals can be judged by comparing them to the range of elevations in each cell. DTED Test Cell #1 contains elevations between 548 m and 2805 m, DTED Test Cell #2 contains elevations between -58 m and 3322 m, DTED Test Cell #3 contains elevations between 40 m and 2915 m, and DTED Test Cell #4 contains elevations between 0 m and 894 m.

The confidence intervals are very different between the four terrain models. This may indicate that a different amount of detail was preserved in the four models, but this difference in preserved detail, if it exists, may be primarily due to the fact that the different terrain areas characterized by these models have very different amounts of original detail that can be preserved. However, this measure of accuracy is of little use in modeling terrain with this specific application of kriging since kriging already assures that for a desired output gridsize, this error variance is minimized. Also, in light of the smoothing discussed above, it is unclear how this type of accuracy measure relates to the quality of the resulting terrain model. More experimentation is needed to fully understand how this error variance relates to desirable characteristics in terrain models.

5.2.3 Kriging from a Partial DTED Cell. In light of the difficulty of kriging based on a full DTED cell and the smoothing resulting from kriging based on filtered DTED, a third possible application of kriging to terrain modeling is to partition the full DTED cell into smaller blocks and to extract new elevation estimates based solely on the data within the same block. This technique should require less memory and may reduce the smoothing due to the density of the elevation data. Partitioning could be performed using `partition` as described in Chapter IV, but for this effort, partitioning was accomplished simply by specifying latitude and longitude boundaries to `dted2pts` when generating the original `.pts` file.

This technique was performed on two different partial cells of DTED. The first is a 10 arc minute latitude by 10 arc minute longitude portion of DTED Test Cell #1 (36°20′N ×112°30′W to 36°30′N ×112°40′W). The second is a 10 arc minute latitude by 10 arc minute longitude portion of DTED Test Cell #2 (38°20′N ×120°30′W to 38°30′N ×120°40′W). These are the same two partial cells used in Section 5.1 above to demonstrate terrain modeling by filtering. Each of these partial cells contain elevation posts arranged on a 201 by 201 grid with grid spacing equal to .05 arc minutes of latitude or longitude (full resolution). Terrain models were generated from these partial cells using `krige` on a grid resolution of 11 by 11 (grid spacing of 0.909 arc minutes of latitude or longitude, roughly equal to the grid spacing of 1 arc minute used by the terrain models generated by the methods above). Larger partial cells proved to contain too large a volume of data to krige successfully; using kriging to build terrain models from these small 10° × 10° blocks took approximately 48 CPU hours on a Silicon Graphics 4D/220.

The experimental variograms and spherical variogram models for these two partial DTED cells are shown in Appendix C. Many of the other full-resolution partial cells studied in this thesis provided more erratic experimental variograms. Although `varfit` tried to fit a spherical variogram model to each of these experimental variograms, many of these models were inappropriate simply because the erratic experimental variogram did not resemble a spherical model. Several of these erratic variograms were actually fitted with invalid models indicated by negative nuggets C_0 or negative ranges a. The two partial cells used in this discussion were chosen because of their well behaved characteristics, better

DTED Test Cell Number	Error Variance	99%Confidence Interval
DTED Test Cell #1	8497.93 m^2	±553.11 m
DTED Test Cell #2	1519.61 m^2	±233.89 m

Table 5 3. Error Variances for Kriged Terrain Models Based on Partial DTED Cells

structural analysis methods need to be developed in order to fully automate the use of these partial cells.

Images rendered from the terrain models built with this technique are shown in Figures 5.21 and 5.22. This model does not seem to show the same smoothing as exhibited in the models referenced above. Several reasons may account for this. First, the elevation posts in the neighborhoods are not filtered, so more real detail remains to generate kriged estimates from. Even though the neighborhood size was restricted by `krige` as for filtered DTED cells, this restriction has less of an effect, for the ranges of the spherical variograms associated with these partial cells are also noticeably smaller. This seems to avocate the use of non-restricted neighborhoods using full resolution DTED, if this were computationally possible. This may also advocate the use of methods that partition DTED cells into zonally isotropic blocks. The partial cells used here may only provide better results because they happen to be more isotropic than a DTED cell as a whole.

Table 5.3 presents the maximum error variances and 99% confidence intervals for these two models These error variances are lower than the error variances of the terrain models produced by kriging based on filtered DTED, indicating that for these small areas the terrain models produced by kriging may be more accurate. This indicates that the error variance may have some usefulness as a measure of accuracy, but again, more experimentation is needed to fully understand how this error variance relates to the quality of the resulting terrain model.

To be useful, these partial cells need to be meshed together into one large terrain model. Although the terrain modeling pipeline can do this, several potential problems exist. First, as mentioned above, `varfit` is not able to generate a valid variogram model for all partial cells. Second, the smaller cells mean that more estimates were made near

the edge of a cell, yielding even less optimal estimates for many of the resulting elevation posts. Finally, the partial cells as presented here may be too small to make it practical to generate a terrain model in this fashion.

5.3 Terrain Modeling using Minimization With Respect To Error Variance.

The final technique tested is the selection of a minimal set of elevation posts from full resolution or filtered DTED based on minimum error variance. The goal of this technique is to provide a way of modeling terrain based on a desired accuracy; this technique allows the user to specify a maximum acceptable error variance instead of a desired resolution. However, this technique did not prove beneficial to terrain modeling, although many of the problems encountered were based on the way this method was implemented.

`krige` uses Brodkin's method of minimization as described in Chapter IV when generating a minimal data set. This method builds the minimal data set as a series of regular patterns. The regularity occurs because new points are always added at the position of maximum error variance, and since the variogram model used to generate the estimates and the error variances is a non-decreasing function, the maximum error variance always occurs at the point the fartherest distance from the points already selected. If the original elevation data is gridded, the patterns are very similar to patterns generated by filtering the DTED. These patterns are filled in with DTED elevation posts until the maximum error variance is below the specified threshold.

In experiments, this error variance threshold seemed to have some relation to the variogram and the resulting density of elevation posts in the minimal data set, but the exact correlation was never apparent. Also, as this threshold is related to the specific variogram of the terrain area under consideration, no "feeling" for the effect of this threshold across different DTED cells could be determined. As stated in Sections 5.2.2 and 5.2.3 above, it seems that more experimentation is needed to fully understand how this error variance relates to the quality of terrain models. No terrain models generated using this minimization technique are presented here due to their similarity to filtered DTED terrain models.

A hypothesis in Chapter III stated that elevation posts extracted from a minimized data set using kriging techniques might characterize the terrain better than any other method. As the minimized data set is almost identical to a filtered data set, a kriged grid of elevation posts from this set would not characterize the original terrain data any better model than a simple filtered version of that data.

5.4 Summary

This chapter presented the results of using the terrain modeling tools and the terrain modeling pipeline presented in Chapter IV to generate accurate and realistic polygonal terrain models by reducing dense terrain elevation data. The basic terrain modeling pipeline did provide some control over the complexity of the resulting terrain model by allowing the user to adjust the DTED sampling interval, but the realism varied with the specific model under consideration and there was no measure of the model's accuracy. The enhanced terrain modeling pipeline encountered several limitations primarily due to memory requirements. In cases where it did work, this system provided a finer level of control over the model's complexity by allowing estimation posts not available in the original DTED to be used; these elevations were estimated using kriging. However, realism suffered due to smoothing and the error variances did not provide adequate measures of accuracy. Using small sections of DTED cells, it was shown that kriging should be able to generate accurate and realistic terrain models if the computational and memory restrictions could be overcome. The technique of minimization with respect to error variance allowed terrain models to be built by controlling the desired accuracy, but such models were built by uniformly increasing the terrain grid resolution to decrease the error variance, and the result was similar to filtering. With no "feel" of how to interpret the error variance threshold, this technique proved of little use. Conclusions based on these observations are presented in Chapter VI.

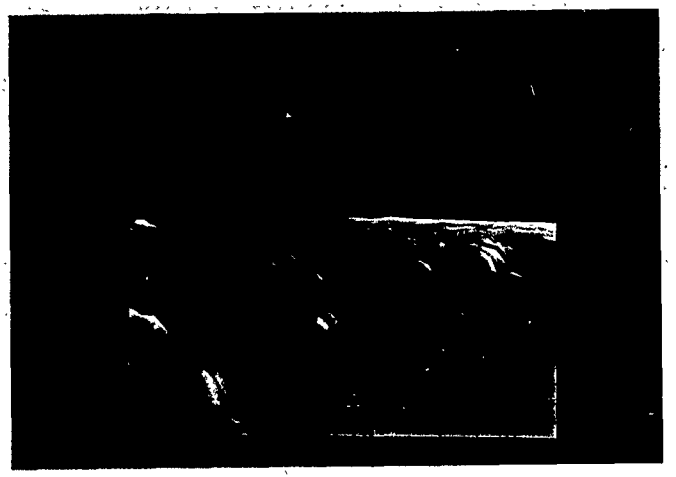


Figure 5.4. Terrain Model Built by Filtering DTED Test Cell #1

Figure 5.5. Terrain Model Built by Filtering DTED Test Cell #2

Figure 5.6. Terrain Model Built by Filtering DTED Test Cell #3

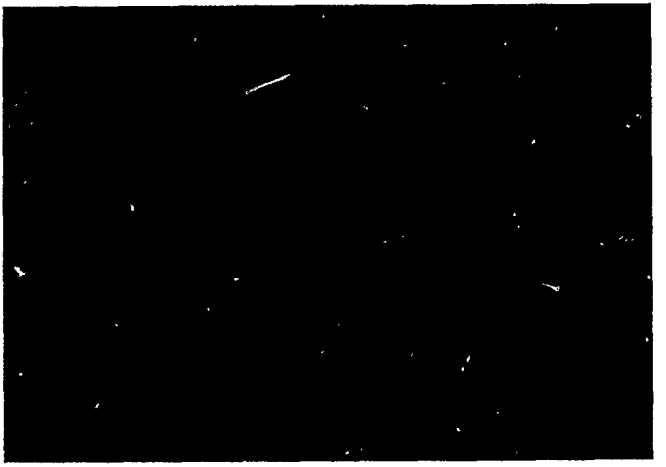


Figure 5.7. Terrain Model Built by Filtering DTED Test Cell #4

Figure 5.8. Terrain Model Built from a 10° × 10° Portion of DTED Test Cell #1 (Full Resolution)

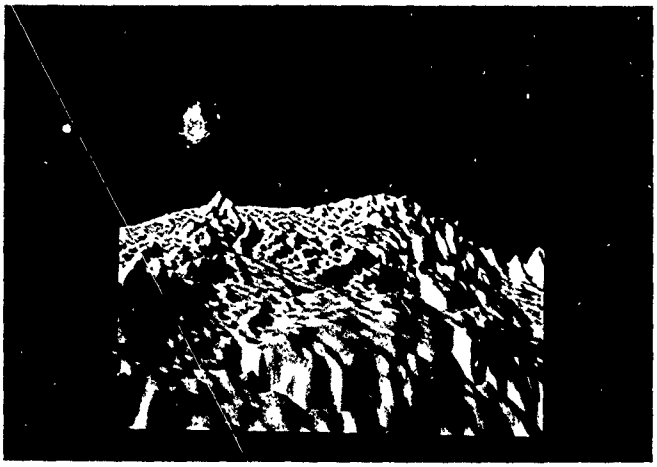


Figure 5.9. Terrain Model Built from a 10° × 10° Portion of DTED Test Cell #2 (Full Resolution)

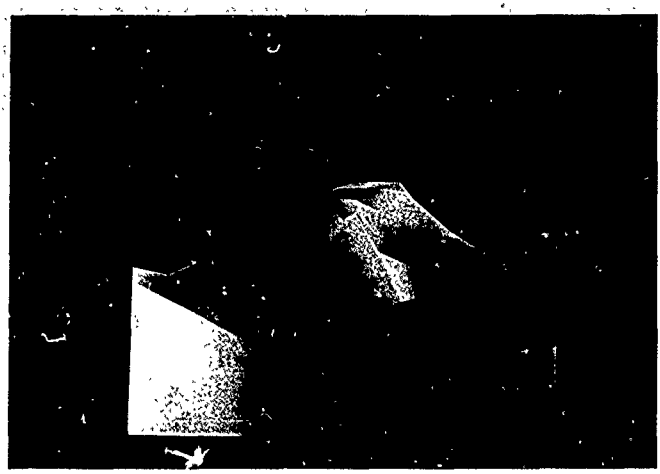


Figure 5.10. Terrain Model Built from a 10° × 10° Portion of DTED Test Cell #1 (Filtered)

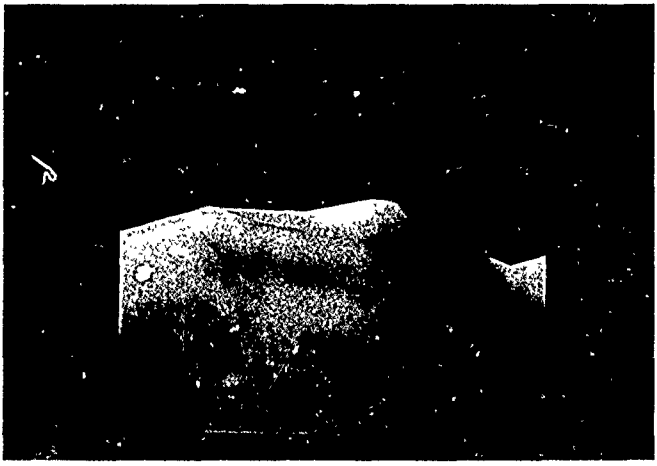


Figure 5.11. Terrain Model Built from a 10° × 10° Portion of DTED Test Cell #2 (Filtered)

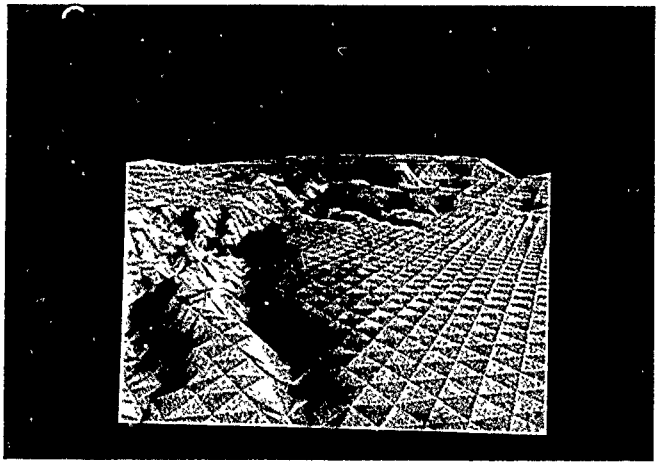


Figure 5.12. Image of Multiple-Level-of-Detail Terrain Model Built from DTED Test Cell #1

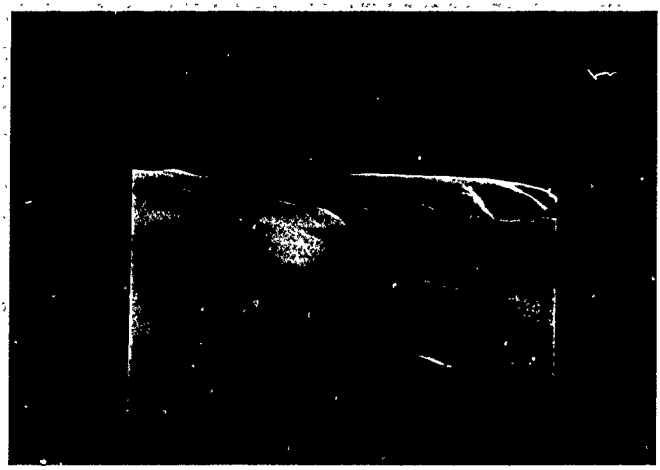


Figure 5.13. Terrain Model Built by Filtering then Kriging DTED Test Cell #1

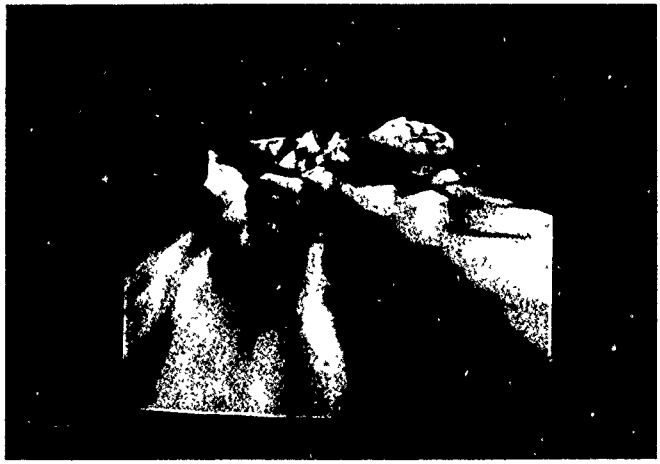


Figure 5.14. Terrain Model Built by Filtering then Kriging DTED Test Cell #2

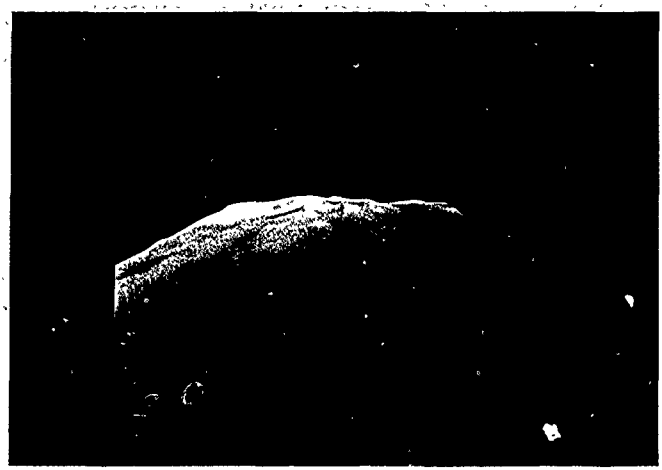


Figure 5.15. Terrain Model Built by Filtering then Kriging DTED Test Cell #3

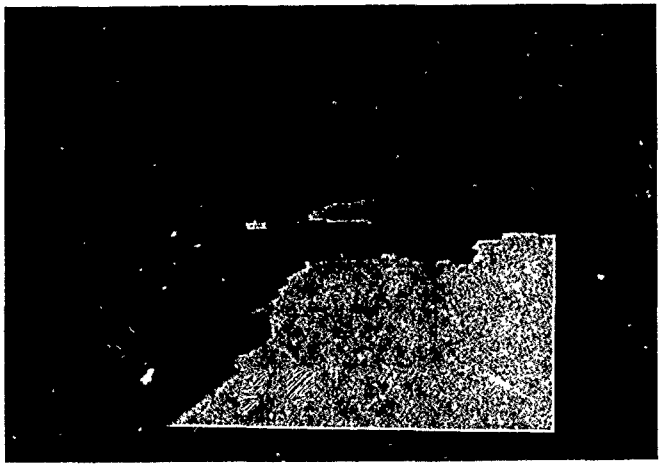


Figure 5.16. Terrain Model Built by Filtering then Kriging DTED Test Cell #4

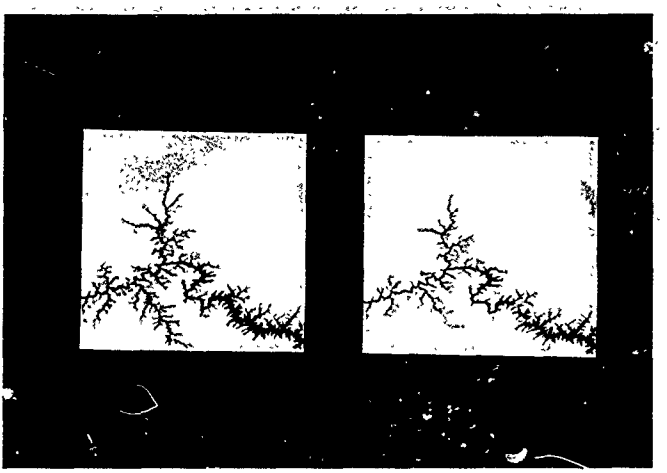


Figure 5.17. Elevation-Shaded 2D Image of Non-Residual (left) and Residual Data (right) from DTED Test Cell #1

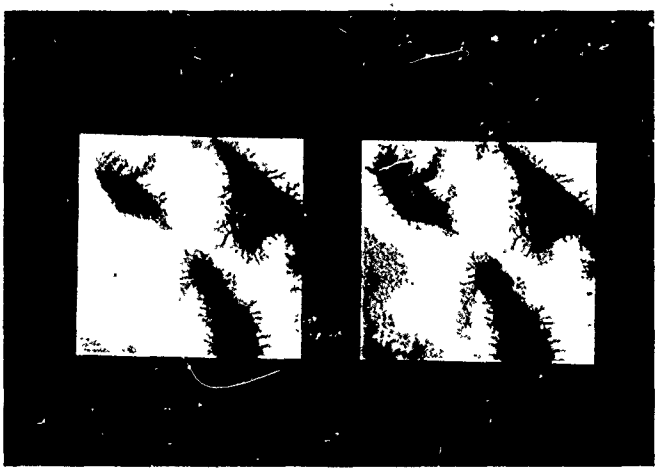


Figure 5 18. Elevation-Shaded 2D Image of Non-Residual (left) and Residual Data (right) from DTED Test Cell #2

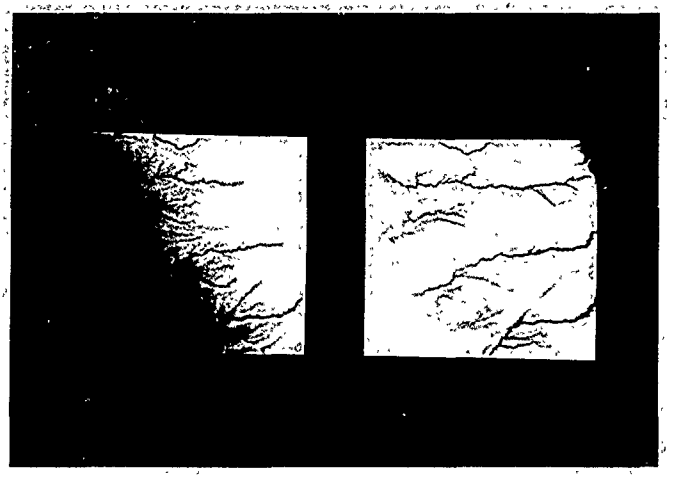


Figure 5.19. Elevation-Shaded 2D Image of Non-Residual (left) and Residual Data (right) from DTED Test Cell #3

Figure 5.20. Elevation-Shaded 2D Image of Non-Residual (left) and Residual Data (right) from DTED Test Cell #4

Figure 5.21. Terrain Model Built by Kriging a 10° × 10° Portion of DTED Test Cell #1 (Full Resolution)

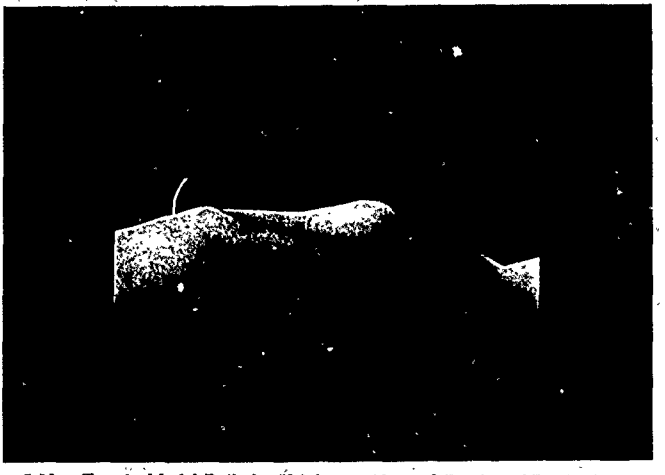


Figure 5.22. Terrain Model Built by Kriging a 10° × 10° Portion of DTED Test Cell #2 (Full Resolution)

VI. Conclusions and Recommendations

This thesis investigates the applicability of kriging as a terrain elevation estimation technique. It also presents the implementation and testing of a terrain modeling system that applies kriging to discrete terrain elevation data such as DTED. This chapter presents conclusions and recommendations based on the results of using this terrain modeling system to reduce dense terrain elevation data in order to build accurate and realistic terrain models.

6.1 Conclusions

Some of the specific conclusions discussed in Chapter V are reiterated below:

- Whole cells of full resolution DTED are the most desirable units of terrain elevation data to use to produce kriged terrain models simply because of the ease of use. However, this type of terrain elevation data provides too great a volume of data to perform kriging estimations using the currently implemented methods.
- The current implementations of the kriging methods can not solve the systems of equations needed to obtain the kriging weights when the kriging neighborhoods contain more than a couple of hundred elevation posts, and even for smaller numbers the time needed to generate an entire kriged model is prohibitive. The techniques used to alleviate these problems also reduced the accuracy of the resulting estimation.
- Kriging as implemented and used in this thesis tends to smooth the terrain elevation data, masking detail that may have been present in the original terrain elevation data.
- The terrain elevation data as used in this thesis may not exhibit the characteristics of a true regionalized variable: the implemented methods of removing global trend may not remove a sufficient amount of such trend to make kriging accurate, and zonal anisotropy within terrain elevation data may also be a problem not sufficiently addressed in this thesis. Both of these problems point to insufficient structural analysis.

- Filtered DTED cells, although containing an amount of data small enough to perform kriging in a practical sense, may not preserve the terrain detail necessary for realistic terrain models. Partial DTED cells are also small enough to successfully krige, but their erratic behavior makes variogram generation difficult.
- The error variance may provide a measure of accuracy in terms of the amount of overall terrain detail preserved, but the measure has no intuitive "feel" across different terrain areas to be useful.
- The minimizing methods implemented achieve no better results than filtering the DTED data, as the error variance measure of accuracy is not well understood for terrain models.

The overall conclusion is that kriging as implemented for this thesis effort does not reduce terrain elevation data well for terrain modeling applications. There is documentation of kriging's effectiveness in minimizing data sets such as Magnetic Resonance Imaging (MRI) scans (2) and in the enhancements of satellite imagery (30); both of these efforts used the same implementation of kriging as presented in this thesis. However, kriging does not perform well for the purpose of reducing large volumes of dense data such as DTED to create accurate and realistic terrain models. This failure seems to be directly related to the volume and density of the DTED and the failure to recognize that the desired result of data reduction – characterizing an area of data by only one point in such a way that the detail in that area is preserved – is fundamentally different than the results obtained from kriging – characterizing the data at any exact point based on analysis of surrounding data.

Kriging may show promise in estimating terrain elevations where the initial terrain elevation data set is considered sparse. Additionally, kriging may prove to be useful in providing elevations between DTED posts for applications that need terrain elevation data at a denser resolution than provided by DTED. Also, such a sparse elevation data set could be generated by hand to build a synthetic terrain surface. A user could place elevation posts only where significant detail is needed, and kriging would estimate other elevation posts on a regular grid based on these few given elevation posts and an assumed variogram. This

technique would also allow models built from such sparse data sets to be easily modified by manipulating the sparse elevation data before kriging; mountains, cliffs, and other features can be moved or scaled easily by moving or scaling selected data points in the sparse data set, and the overall "roughness" may be variable using different variograms.

6.2 Recommendations

As noted above, several problems arose during kriging that are associated with either the density or the volume of the DTED used or the inaccuracies resulting from the reduction of the DTED using kriging. Listed below are several recommended modifications to the kriging estimation methods implemented in this thesis that may improve the performance and/or the results of terrain modeling using kriging.

- `krige` is a natural candidate for parallel processing. If n elevation estimates need to be made, n separate systems of equations need to be solved and n weighted sums need to be calculated. `krige` currently performs these iterations using a loop construct, so the re-implementation into parallel code could be easily performed.
- The technique of dual kriging, briefly mentioned in Chapter II, avoids the time consuming step of solving for the kriging weights for each estimate by modeling the weights as a function of distance. This technique should greatly reduce the time necessary to determine a kriged estimate.
- As the technique of matrix inversion using LU decomposition and backsubstitution becomes unstable for large matrices, other techniques need to be considered. One possible technique, although not necessarily faster, is to iteratively apply the *inversion by blocks* technique presented in Chapter III, starting with a small portion of the matrix and eventually inverting the entire matrix.
- In general, the structural analysis processing done by `resid` and `varfit` needs to be more robust. Since `resid` does not seem to remove a sufficient amount of global trend, perhaps a different method of removing trend needs to be explored or a different type of trend needs to be removed. In the later case, trend based on a periodic function may remove more trend from terrain data than trend based on a polynomial. Also,

as noted in the conclusions above, the variogram model generated by `varfit` may not correspond with the actual variogram very well; this may be improved with a better model fitting algorithm, or even by generating models variograms by hand.

- As mentioned in the conclusions, partitioning the terrain elevation data would help to ensure zonal isotropy, if the partitioning is performed properly. Certainly the use of the `partition` program needs to be explored farther. However, another possible technique to partition an area of residual terrain elevation data is to divide the subject area into quads and generate directional variograms on each quad; if any of the directional variograms are dissimilar to the corresponding directional variograms generated from the parent area, repeat the division and variogram generation process on this particular quad. Continue this until the directional variograms of the four quads of an area are similar, then re-combine these quads into one area. As this technique uses the actual variogram as a measure of anisotropy, this should yield a patchwork of sub-areas that are each zonal isotropic.

- Another potential partitioning method requires the generation of mini-variograms over a series of small windows in the terrain data set. These variograms should be analyzed for significant changes; the window position where a significant variogram change takes place should indicate a position where the terrain characteristics change significantly. The terrain can then be partitioned along boundaries determined by this method before kriging is performed. However, a better understanding of the characteristics of a variogram model and their relationship to terrain texture and detail is needed before this method can be fully implemented.

- Chapter III describes several methods for selecting a minimal subset of an original data set such that the maximum error variance is also minimized. As noted above, Brodkin's method yields a distinct pattern of data points that have no correlation with the areas of significant detail. Szidarovszky's methods presented in Chapter III for selecting such a minimal data set should be tried to determine if they have the same shortcoming.

- All of the minimization methods reviewed in Chapter III and above use the error variance as the measure of accuracy. As stated in the conclusions above, this does

not necessarily correspond to accuracy in terms of preserving specific details of the terrain. However, an alternate method of selecting a minimal data set has been implemented into `krige` but has not been evaluated. This method is similar to the method of selecting points with the maximum error variance, but it uses the criteria of selecting points with the *maximum actual error* between their estimated and real values. Since actual error is not solely dependent on distances, the points selected would not necessarily fall into a pattern; therefore each pass must consider the entire surface to add just one point. This method would be more time-intensive, but should yield a pattern of elevation posts that exhibit more of the detail of the terrain surface than the other methods considered. It is interesting to note that this method works because of the inaccuracies of the kriging methods as noted above; data points are added at positions where the kriged estimate exhibits these inaccuracies the most.

As this data set would not necessarily be aligned in a regular grid, two approaches could be taken to create a terrain model: the first would be to extract a gridded data set from this irregular data set using `krige`, although this would tend to smooth the detail that one was attempting to preserve by building this minimal data set. The second alternative is to implement a irregular triangular meshing algorithm as referenced in Chapter II; this should create a model with all the detail present in the minimal data set.

- Finally, a synthetic terrain model could easily be generated from sparse elevation data laid out by hand and then kriged to a specific gird. This application of kriging would avoid most of the problems encountered while kriging DTED data due to the sparse nature of the original data used and the fact that this data by definition reflects the significant details of the desired terrain surface.

6.3 Summary

This chapter presented the conclusions and recommendations based on this thesis research effort. Based on the results in Chapter V, the overall conclusion is that kriging as currently implemented does not reduce terrain elevation data well for terrain modeling purposes, but modifications to the kriging methods as recommended above may provide

improvements in the performance and results of this terrain modeling technique. Kriging needs to be explored further to determine if the methods implemented in this thesis or their methods of application can be modified to produce acceptable polygonal terrain models.

Appendix A. *Development of Kriging Equations*

An overview of the kriging equations are presented in Chapter II. A more complete development of these equations, as adapted from Journel (25), follow.

The intent of kriging is to provide a weighted-sum estimate $Z^*(s_0)$ of the spatial property at an unknown position s_0 using Eq.(A.1), which is a slightly modified version of Eq (2.1):

$$\begin{aligned} Z^*(s_0) &= w_0 + w_1 Z(s_1) + w_2 Z(s_2) + w_3 Z(s_3) + \cdots + w_n Z(s_n) \\ &= w_0 + \sum_{i=1}^{n} w_i Z(s_i) \end{aligned} \tag{A.1}$$

where s_i, $0 < i < n$, refers to n positions on the regionalized variable where the value of the spatial property is known; $Z(s_i)$ refers to the value of the spatial property at position s_i; w_i, $0 < i < n$, are weights applied to the sampled values; and w_0 is known as a shift factor.

The weights w_i in Eq (A.1) should be selected such that the estimator is the *best* estimator according to some measure. Since little can be determined about the actual estimation error $Z(s_0) - Z^*(s_0)$ without a foreknowledge of $Z(s_0)$, kriging defines the *best* estimator as the estimator with a zero error mean μ_E and the minimum error variance σ_E^2. The error mean μ_E is calculated as follows:

$$\begin{aligned} \mu_E &= E[Z(s_0) - Z^*(s_0)] \\ &= E[Z(s_0)] - E[Z^*(s_0)] \\ &= E[Z(s_0)] - E[w_0 + \sum_{i=1}^{n} w_i Z(s_i)] \\ &= E[Z(s_0)] - w_0 - \sum_{i=1}^{n} w_i E[Z(s_i)] \\ &= \mu - w_0 - \sum_{i=1}^{n} w_i \mu \\ &= -w_0 + \mu \left(1 - \sum_{i=1}^{n} w_i\right) \end{aligned} \tag{A.2}$$

To ensure this value equals zero no matter the value of μ, the following conditions must be true:

$$\begin{aligned} w_0 &= 0 \\ \textstyle\sum_{i=1}^{n} w_i &= 1 \end{aligned} \tag{A.3}$$

Substituting into Eq (A.1), the resulting estimator for $Z(s_0)$ is simply:

$$Z^*(s_0) = \sum_{i=1}^{n} w_i Z(s_i) \tag{A.4}$$

This equation is identical to Eq (2.1).

The condition that the weights sum to one, as shown in Eq (A.3), is traditionally known as the unbiasedness condition. This condition requires that, "on average, the value which is computed, should be equal to the real value, rather than systematically higher or lower" (11:238).

In order to minimize the error variance, the actual error $Z(s_0) - Z^*(s_0)$, although unknown, can be written using Eq (A.4) above:

$$Z(s_0) - Z^*(s_0) = Z(s_0) - \sum_{i=1}^{n} w_i Z(s_i) \tag{A.5}$$

Using the notation of

$$\begin{aligned} a_0 &= 1 \\ a_i &= -w_i \quad i = 1, .., n \end{aligned} \tag{A.6}$$

Eq (A.5) can be rewritten as:

$$\begin{aligned} Z(s_0) - Z^*(s_0) &= a_0 Z(s_0) - \sum_{i=1}^{n} -a_i Z(s_i) \\ &= a_0 Z(s_0) + \sum_{i=1}^{n} a_i Z(s_i) \\ &= \sum_{i=0}^{n} a_i Z(s_i) \end{aligned} \tag{A.7}$$

Using this equation, the error variance can be written as:

$$\sigma_E^2 = Var[Z(s_0) - Z^*(s_0)]$$

$$= Var[\sum_{i=0}^{n} a_i Z(s_i)]$$
$$= \sum_{i=0}^{n}\sum_{j=0}^{n} a_i a_j \sigma_{ij} \qquad \text{(A.8)}$$

where $\sigma_{ij} = \text{Cov}[Z(s_i), Z(s_j)]$.

To minimize this equation, the partial derivatives with respect to each of the coefficients a_i, $0 < i \leq n$ must be set to zero:

$$\frac{\partial(\sigma_E^2)}{\partial a_i} = 0, \quad i = 1, ..., n \qquad \text{(A.9)}$$

Since $a_0 = 0$ is known from above, the partial derivative with respect to this coefficient is not taken; and since $a_1 = -w_i$, $0 < i \leq n$, the resulting equation is identical to Eq (2.4), as shown below:

$$\frac{\partial(\sigma_E^2)}{\partial a_i} = -\frac{\partial(\sigma_E^2)}{\partial w_i} = \frac{\partial(\sigma_E^2)}{\partial w_i} = 0 \quad i = 1, ..., n \qquad \text{(A.10)}$$

Using Eq (A.8), the partial derivatives of σ_E^2 with respect to the coefficients a_i are of the form:

$$\frac{\partial(\sigma_E^2)}{\partial a_i} = 2\sum_{j=0}^{n} a_j \sigma_{ij}$$
$$= 2a_0\sigma_{i0} + 2\sum_{j=1}^{n} a_j \sigma_{ij}$$

and substituting $w_i = a_i$, $0 < i \leq n$, and $a_0 = 1$:

$$\frac{\partial(\sigma_E^2)}{\partial w_i} = 2\sigma_{i0} - 2\sum_{j=1}^{n} w_j \sigma_{ij} \qquad \text{(A.11)}$$

In order to minimize the error variance, these partial derivatives must be set to zero:

$$2\sigma_{i0} - 2\sum_{j=1}^{n} w_j \sigma_{ij} = 0$$
$$\sum_{j=1}^{n} w_j \sigma_{ij} = \sigma_{i0} \qquad \text{(A.12)}$$

When this sum is expanded for each w_i a system of n equations with n unknowns results. When the unbiasedness condition of Eq (A.3) is included in the system of equations, a LaGrange multiplier λ is added to sufficiently constrain the system (12:385). This entire system of equations, originally provided in Eq (2.5), is shown below:

$$\begin{array}{ccccccccccc}
w_1\sigma_{11} & + & w_2\sigma_{12} & + & \cdots & + & w_n\sigma_{1n} & + & \lambda & = & \sigma_{10} \\
w_1\sigma_{21} & + & w_2\sigma_{22} & + & \cdots & + & w_n\sigma_{2n} & + & \lambda & = & \sigma_{20} \\
w_1\sigma_{31} & + & w_2\sigma_{32} & + & \cdots & + & w_n\sigma_{3n} & + & \lambda & = & \sigma_{30} \\
\cdots & + & \cdots & + & \cdots & + & \cdots & + & \cdots & = & \cdots \\
w_1\sigma_{n1} & + & w_2\sigma_{n2} & + & \cdots & + & w_n\sigma_{nn} & + & \lambda & = & \sigma_{n0} \\
w_1 & + & w_2 & + & \cdots & + & w_n & & & = & 1
\end{array} \tag{A.13}$$

As shown in Chapter II, this system of equations can be solved as a matrix equation of the form:

$$Aw = B \tag{A.14}$$

where A, w, and B refer to the following matrices:

$$A = \begin{bmatrix}
\sigma_{11} & \sigma_{12} & \cdots & \sigma_{1n} & 1 \\
\sigma_{21} & \sigma_{22} & \cdots & \sigma_{2n} & 1 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
\sigma_{n1} & \sigma_{n2} & \cdots & \sigma_{nn} & 1 \\
1 & 1 & \cdots & 1 & 0
\end{bmatrix} \tag{A.15}$$

$$w^T = \begin{bmatrix} w_1 & w_2 & \cdots & w_n & \lambda \end{bmatrix} \tag{A.16}$$

$$B^T = \begin{bmatrix} \sigma_{10} & \sigma_{20} & \cdots & \sigma_{n0} & 1 \end{bmatrix} \tag{A.17}$$

The weights w_i are determined by:

$$w = A^{-1}B \tag{A.18}$$

After the weights are calculated, Eq (A.4) can be used to calculate an estimate $Z^*(s_0)$ for location s_0. Even though λ is necessary in the system of equations, its value is unimportant to the value of the estimate.

To determine the resulting error variance σ_E^2 of this estimate, Eq (A.8) can be further decomposed as follows:

$$\begin{aligned} \sigma_E^2 &= \sum_{i=0}^{n}\sum_{j=0}^{n} a_i a_j \sigma_{ij} \\ &= \underbrace{\sigma_{00}}_{i,j=0} - \underbrace{\sum_{j=1}^{n} a_j \sigma_{0j}}_{i=0,j>0} - \underbrace{\sum_{i=1}^{n} a_i \sigma_{i0}}_{i>0,j=0} + \underbrace{\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j \sigma_{ij}}_{i,j>0} \\ &= \sigma_{00} - 2\sum_{i=1}^{n} a_i \sigma_{i0} + \sum_{i=1}^{n} a_i \left(\sum_{j=1}^{n} a_j \sigma_{ij} \right) \end{aligned} \tag{A.19}$$

Using Eq (A.12) and the fact that $\sigma_{00} = \sigma_{ii} = \sigma^2$, the error variance can be rewritten as:

$$\sigma_E^2 = \sigma^2 - \sum_{i=1}^{n} a_i \sigma_{i0} \tag{A.20}$$

or, substituting the weights w_i from Eq (A.6) back in:

$$\sigma_E^2 = \sigma^2 - \sum_{i=1}^{n} w_i \sigma_{i0} \tag{A.21}$$

This also results in a system of n equations with n unknowns similar to those resulting from Eq (A.12). Once the unbiasedness condition and LaGrange multiplier are introduced,

a system of equations are generated of the form:

$$\sigma_E^2 = \sigma^2 - w^T B \tag{A.22}$$

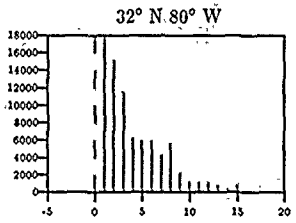
where w and B are the same matrices as in Eqs (A.16) and (A.17) above.

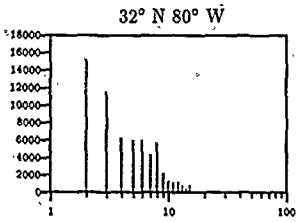
In all cases above, it is assumed that σ^2 and σ_{ij} for any i and j can be determined from the variogram.

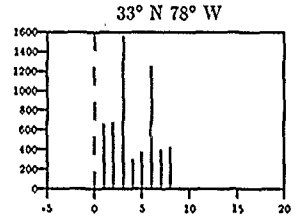
Appendix B. *Terrain Elevation Distributions*

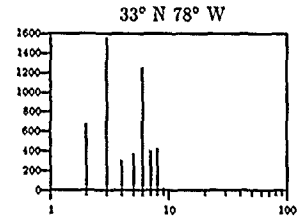
B.1 Distributions of Raw Terrain Elevation Data

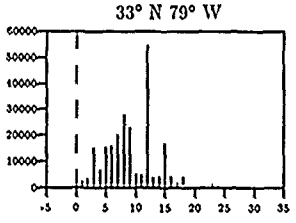
The following pages show terrain elevation histograms from full-resolution DMA DTED as taken from the DTED Evaluation Cells listed in Chapter III. The latitude and longitude of the southwest corner of the DTED cell used to build each histogram is provided above the histogram. All histograms are shown on both normal and log-normal scales. If a particular cell contains zero-elevation data, this data has been removed before the histograms were generated; such data often covers a large portion of the cell and therefore causes data at other elevations to be insignificant.

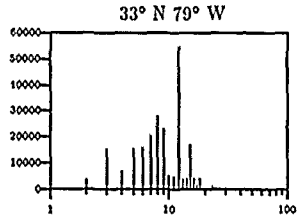

32° N 80° W
18000
16000
14000
12000
10000
8000
6000
4000
2000
0
-5
0
5
10
15
20

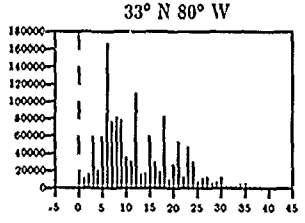

32° N 80° W
18000
16000
14000
12000
10000
8000
6000
4000
2000
0
1
10
100

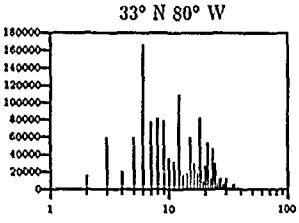

33° N 78° W
1600
1400
1200
1000
800
600
400
200
0
-5
0
5
10
15
20


33° N 78° W
1600
1400
1200
1000
800
600
400
200
0
1
10
100

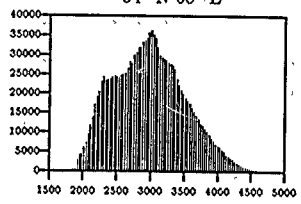

33° N 79° W
60000
50000
40000
30000
20000
10000
0
-5
0
5
10
15
20
25
30
35


33° N 79° W
60000
50000
40000
30000
20000
10000
0
1
10
100

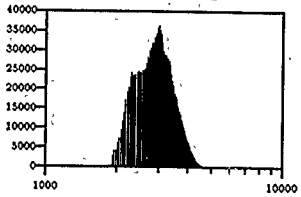

33° N 80° W
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
-5
0
5
10
15
20
25
30
35
40
45


33° N 80° W
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
1
10
100

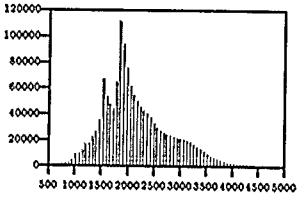
34° N 68° E



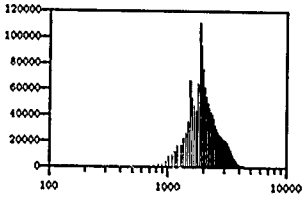
34° N 68° E



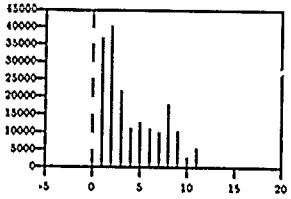
34° N 69° E



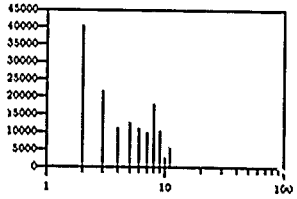
34° N 69° E



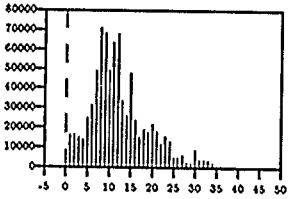
34° N 77° W



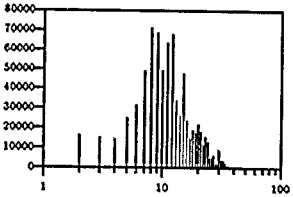
34° N 77° W

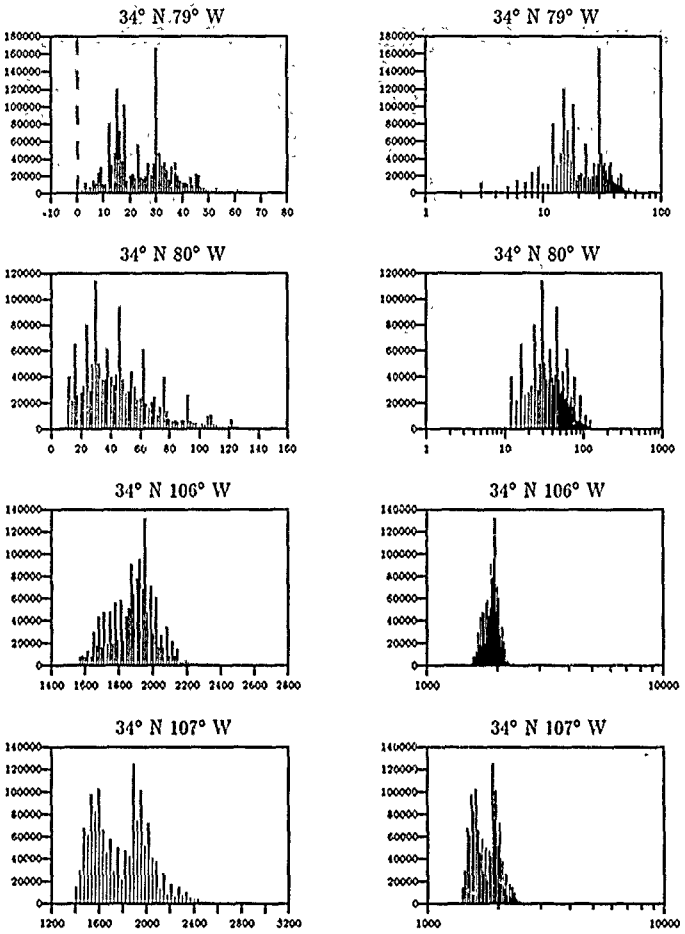


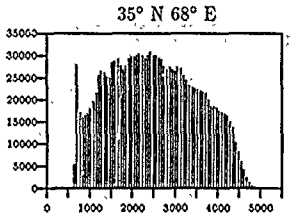
34° N 78° W

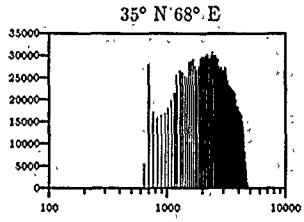


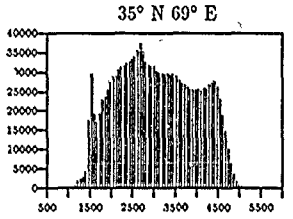
34° N 78° W

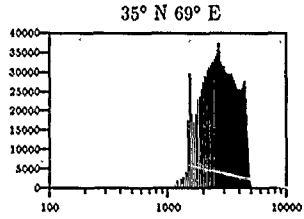


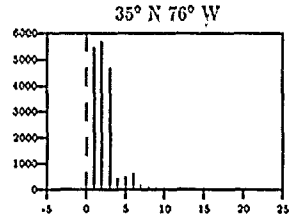

34° N 79° W
34° N 79° W
34° N 80° W
34° N 80° W
34° N 106° W
34° N 106° W
34° N 107° W
34° N 107° W

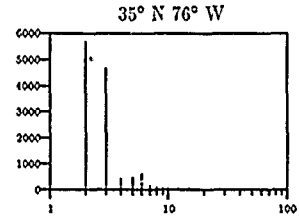

35° N 68° E
35000
30000
25000
20000
15000
10000
5000
0
0 1000 2000 3000 4000 5000

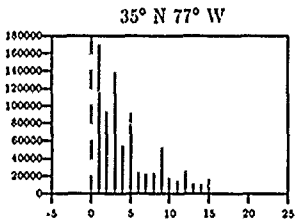

35° N 68° E
35000
30000
25000
20000
15000
10000
5000
0
100 1000 10000

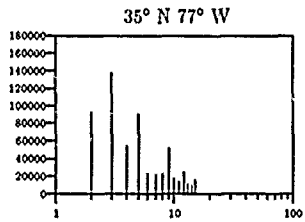

35° N 69° E
40000
35000
30000
25000
20000
15000
10000
5000
0
500 1500 2500 3500 4500 5500

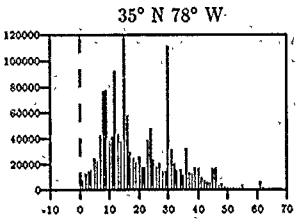

35° N 69° E
40000
35000
30000
25000
20000
15000
10000
5000
0
100 1000 10000

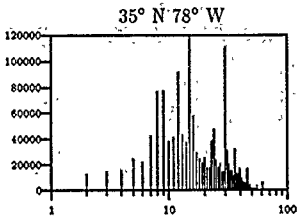

35° N 76° W
6000
5000
4000
3000
2000
1000
0
-5 0 5 10 15 20 25

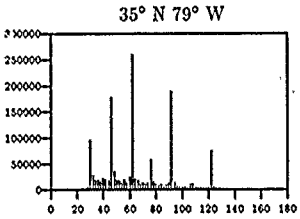

35° N 76° W
6000
5000
4000
3000
2000
1000
0
1 10 100

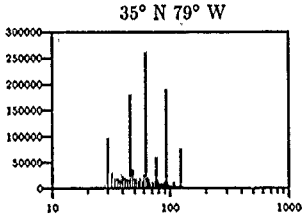

35° N 77° W
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
-5 0 5 10 15 20 25

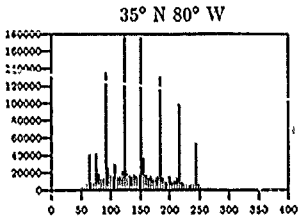

35° N 77° W
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
1 10 100

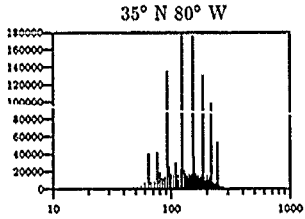

35° N 78° W
120000
100000
80000
60000
40000
20000
0
-10 0 10 20 30 40 50 60 70

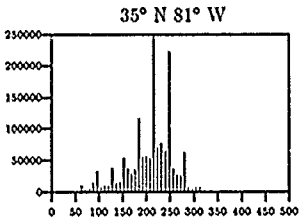

35° N 78° W
120000
100000
80000
60000
40000
20000
0
1 10 100

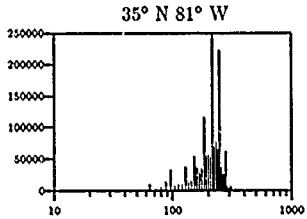

35° N 79° W
300000
250000
200000
150000
100000
50000
0
0 20 40 60 80 100 120 140 160 180

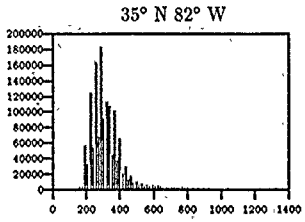

35° N 79° W
300000
250000
200000
150000
100000
50000
0
10 100 1000

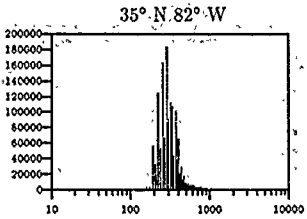

35° N 80° W
160000
140000
120000
100000
80000
60000
40000
20000
0
0 50 100 150 200 250 300 350 400

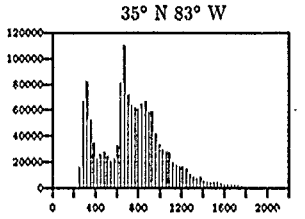

35° N 80° W
180000
160000
140000
120000
80000
60000
40000
20000
0
10 100 1000

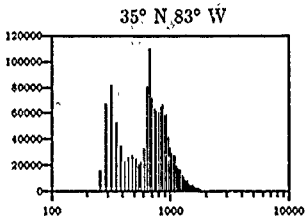

35° N 81° W
250000
200000
150000
100000
50000
0
0 50 100 150 200 250 300 350 400 450 500

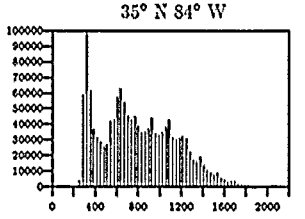

35° N 81° W
250000
200000
150000
100000
50000
0
10 100 1000

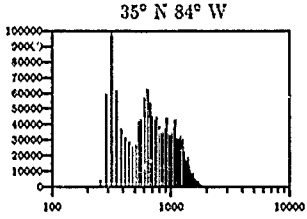

35° N 82° W
200000
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
0 200 400 600 800 1000 1200 1400

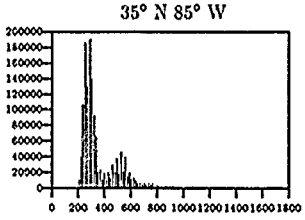

35° N 82° W
200000
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
10 100 1000 10000

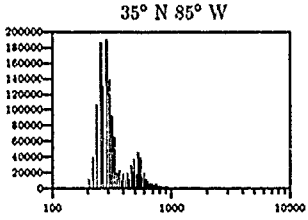

35° N 83° W
120000
100000
80000
60000
40000
20000
0
0 400 800 1200 1600 2000

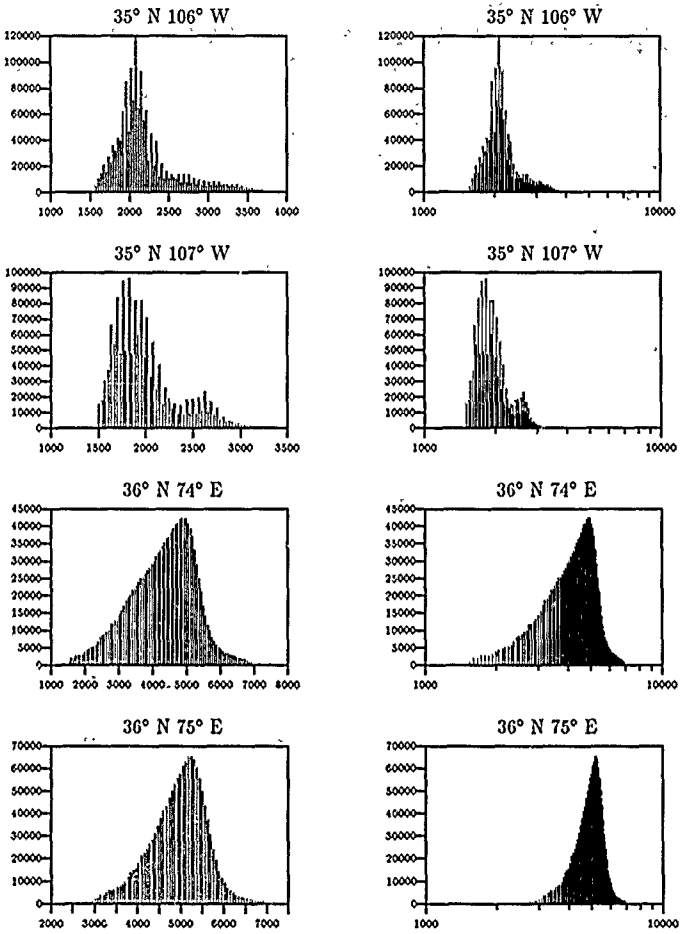

35° N 83° W
120000
100000
80000
60000
40000
20000
0
100 1000 10000

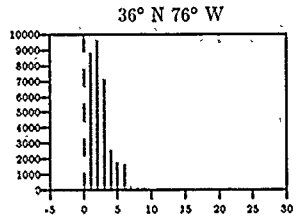

35° N 84° W
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
0 400 800 1200 1600 2000

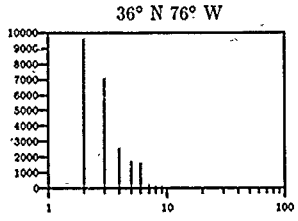

35° N 84° W
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
100 1000 10000

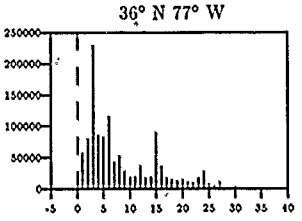

35° N 85° W
200000
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
0 200 400 600 800 1000 1200 1400 1600 1800

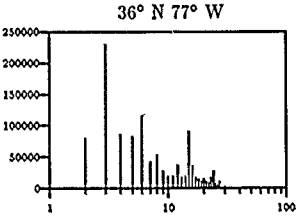

35° N 85° W
200000
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
100 1000 10000

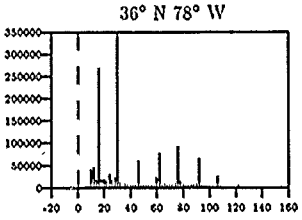

35° N 106° W
35° N 106° W
35° N 107° W
35° N 107° W
36° N 74° E
36° N 74° E
36° N 75° E
36° N 75° E

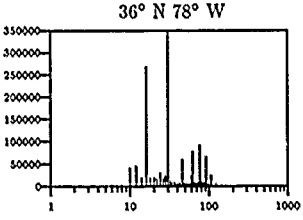

36° N 76° W
10000
9000
8000
7000
6000
5000
4000
3000
2000
1000
0
-5 0 5 10 15 20 25 30

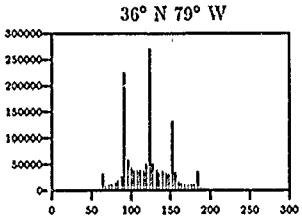

36° N 76° W
10000
9000
8000
7000
6000
5000
4000
3000
2000
1000
0
1 10 100

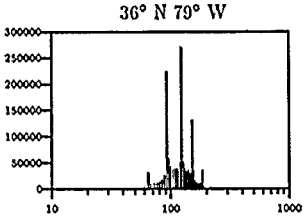

36° N 77° W
250000
200000
150000
100000
50000
0
-5 0 5 10 15 20 25 30 35 40

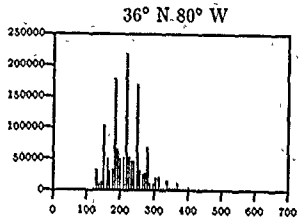

36° N 77° W
250000
200000
150000
100000
50000
0
1 10 100

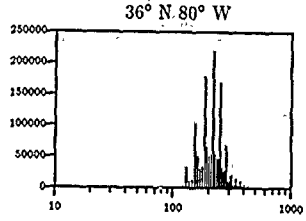

36° N 78° W
350000
300000
250000
200000
150000
100000
50000
0
-20 0 20 40 60 80 100 120 140 160

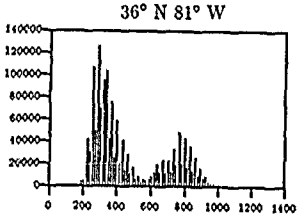

36° N 78° W
350000
300000
250000
200000
150000
100000
50000
0
1 10 100 1000

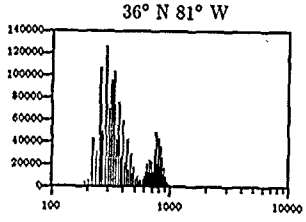

36° N 79° W
300000
250000
200000
150000
100000
50000
0
0 50 100 150 200 250 300

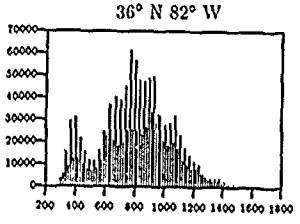

36° N 79° W
300000
250000
200000
150000
100000
50000
0
10 100 1000

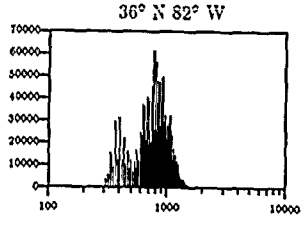

36° N 80° W
250000
200000
150000
100000
50000
0
0 100 200 300 400 500 600 700

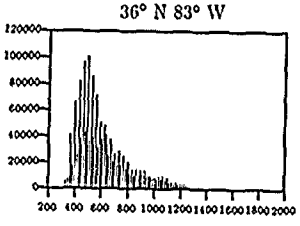

36° N 80° W
250000
200000
150000
100000
50000
0
10 100 1000

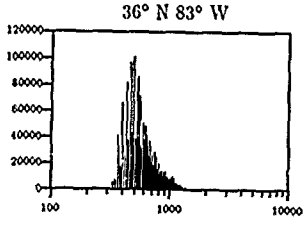

36° N 81° W
140000
120000
100000
80000
60000
40000
20000
0
0 200 400 600 800 1000 1200 1400

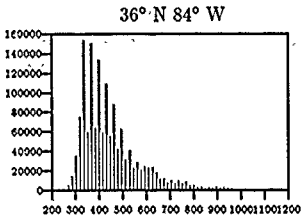

36° N 81° W
140000
120000
100000
80000
60000
40000
20000
0
100 1000 10000

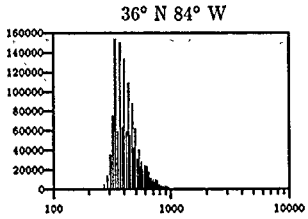

36° N 82° W
70000
60000
50000
40000
30000
20000
10000
0
200 400 600 800 1000 1200 1400 1600 1800

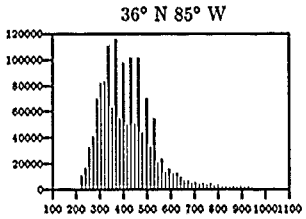

36° N 82° W
70000
60000
50000
40000
30000
20000
10000
0
100 1000 10000

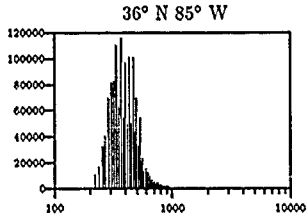

36° N 83° W
120000
100000
80000
60000
40000
20000
0
200 400 600 800 1000 1200 1400 1600 1800 2000

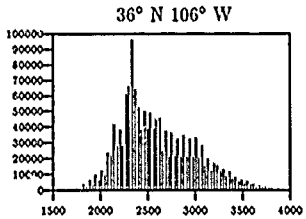

36° N 83° W
120000
100000
80000
60000
40000
20000
0
100 1000 10000

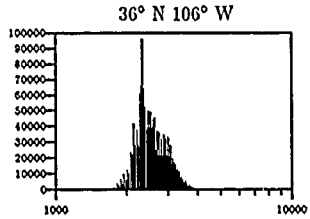

36° N 84° W
160000
140000
120000
100000
80000
60000
40000
20000
0
200 300 400 500 600 700 800 900 1000 1100 1200

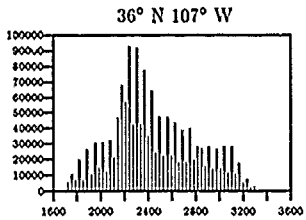

36° N 84° W
160000
140000
120000
100000
80000
60000
40000
20000
0
100
1000
10000

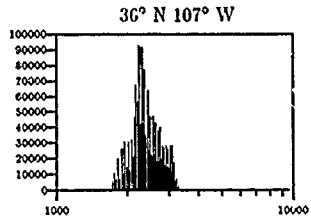

36° N 85° W
120000
100000
80000
60000
40000
20000
0
100 200 300 400 500 600 700 800 900 1000 1100

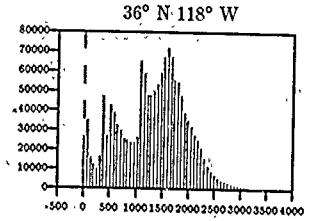

36° N 85° W
120000
100000
80000
60000
40000
20000
0
100
1000
10000

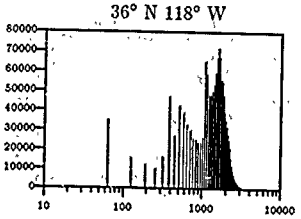

36° N 106° W
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
1500 2000 2500 3000 3500 4000

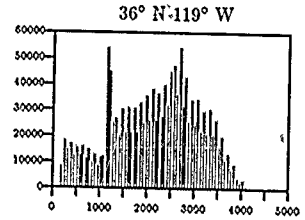

36° N 106° W
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
1000
10000

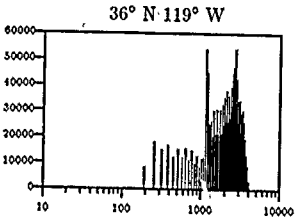

36° N 107° W
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
1600 2000 2400 2800 3200 3600

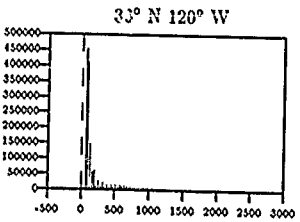

36° N 107° W
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
1000
10000

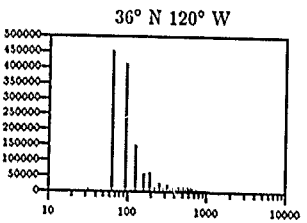

36° N 118° W
80000
70000
60000
50000
40000
30000
20000
10000
0
-500 0 500 1000 1500 2000 2500 3000 3500 4000

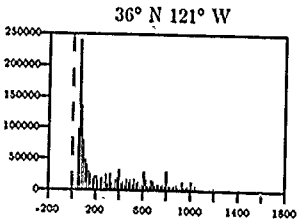

36° N 118° W
80000
70000
60000
50000
40000
30000
20000
10000
0
10 100 1000 10000

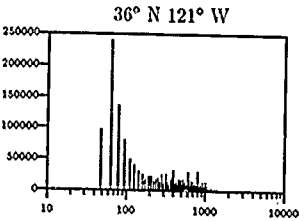

36° N 119° W
60000
50000
40000
30000
20000
10000
0
0 1000 2000 3000 4000 5000


36° N 119° W
60000
50000
40000
30000
20000
10000
0
10 100 1000 10000

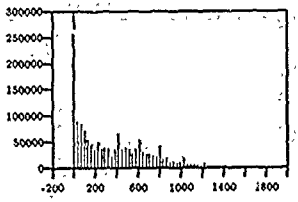

36° N 120° W
500000
450000
400000
350000
300000
250000
200000
150000
100000
50000
0
-500 0 500 1000 1500 2000 2500 3000


36° N 120° W
500000
450000
400000
350000
300000
250000
200000
150000
100000
50000
0
10 100 1000 10000

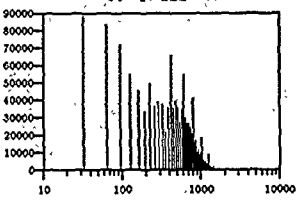

36° N 121° W
250000
200000
150000
100000
50000
0
-200 200 600 1000 1400 1800


36° N 121° W
250000
200000
150000
100000
50000
0
10 100 1000 10000

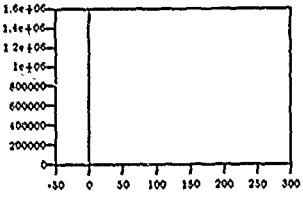
36° N 122° W



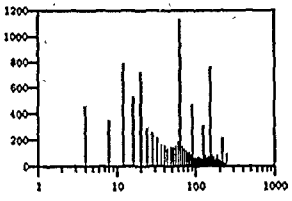
36° N 122° W



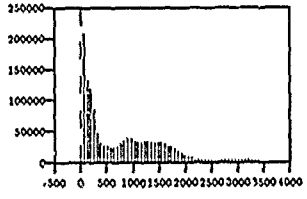
36° N 123° W



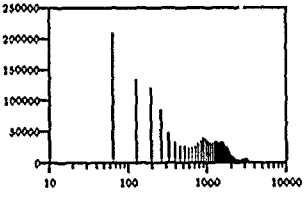
36° N 123° W



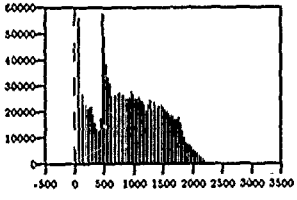
37° N 35° E



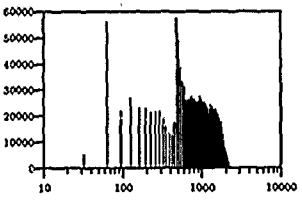
37° N 35° E



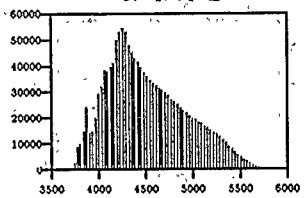
37° N 36° E



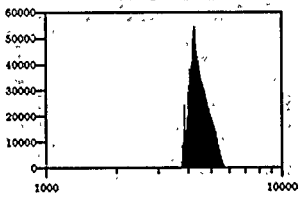
37° N 36° E



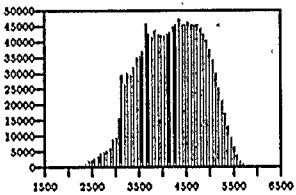
37° N 74° E



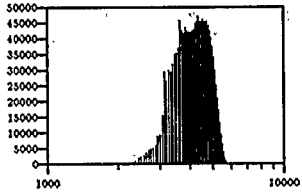
37° N 74° E



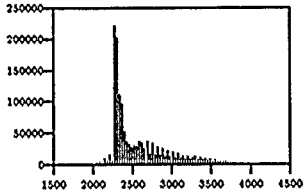
37° N 75° E



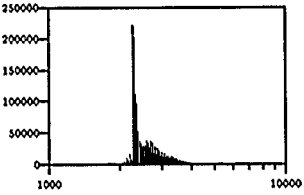
37° N 75° E



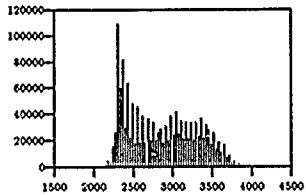
37° N 106° W



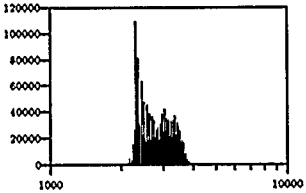
37° N 106° W

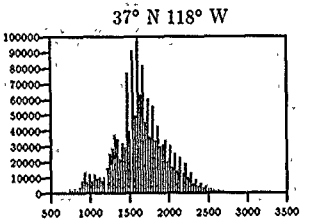


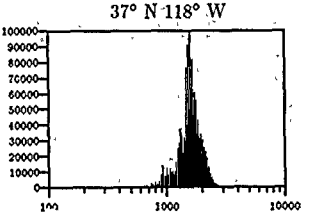
37° N 107° W

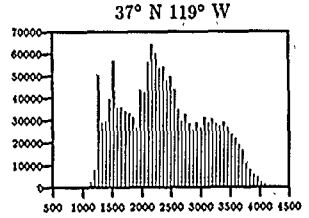


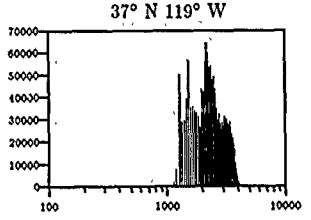
37° N 107° W

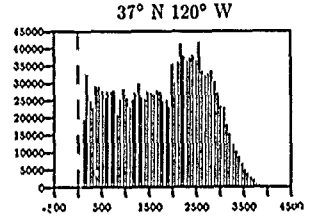


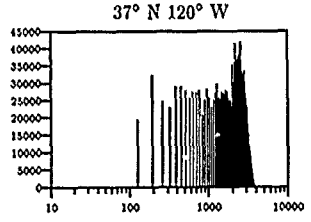

37° N 118° W
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
500 1000 1500 2000 2500 3000 3500

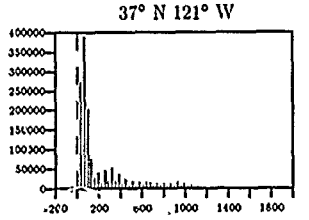

37° N 118° W
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
100 1000 10000

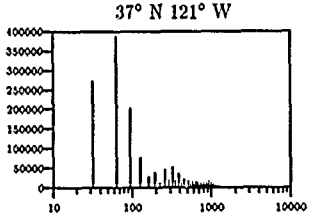

37° N 119° W
70000
60000
50000
40000
30000
20000
10000
0
500 1000 1500 2000 2500 3000 3500 4000 4500


37° N 119° W
70000
60000
50000
40000
30000
20000
10000
0
100 1000 10000

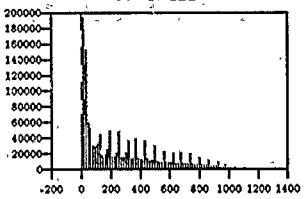

37° N 120° W
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
500 1500 2500 3500 4500


37° N 120° W
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
10 100 1000 10000

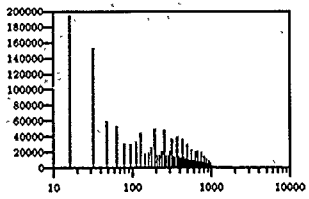

37° N 121° W
400000
350000
300000
250000
200000
150000
100000
50000
0
200 600 1000 1400 1800


37° N 121° W
400000
350000
300000
250000
200000
150000
100000
50000
0
10 100 1000 10000

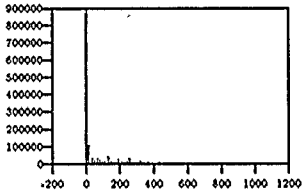
37° N 122° W



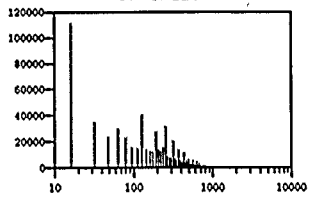
37° N 122° W



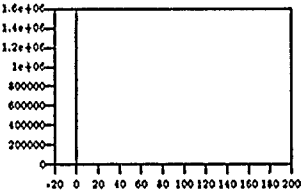
37° N 123° W



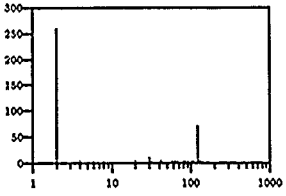
37° N 123° W



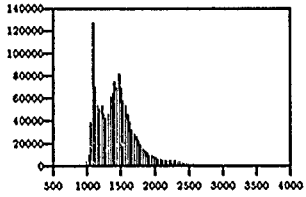
37° N 124° W



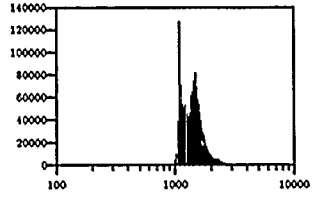
37° N 124° W

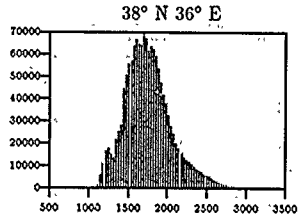


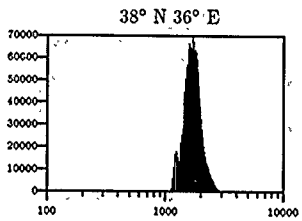
38° N 35° E

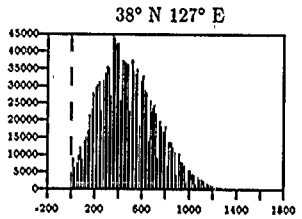


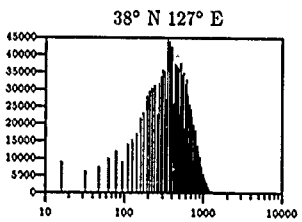
38° N 35° E

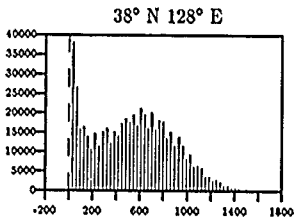


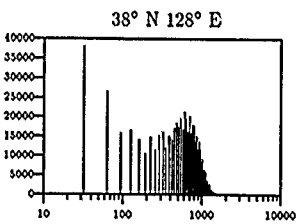

38° N 36° E
70000
60000
50000
40000
30000
20000
10000
0
500 1000 1500 2000 2500 3000 3500

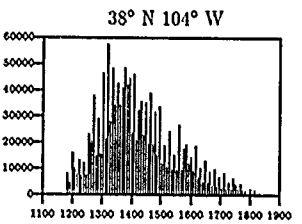

38° N 36° E
70000
60000
50000
40000
30000
20000
10000
0
100 1000 10000

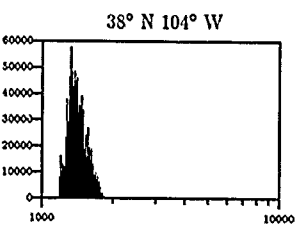

38° N 127° E
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
-200 200 600 1000 1400 1800

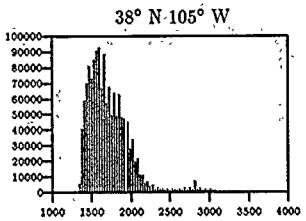

38° N 127° E
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
10 100 1000 10000

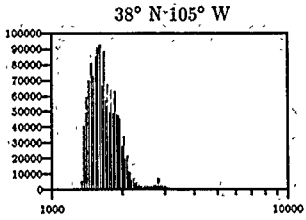

38° N 128° E
40000
35000
30000
25000
20000
15000
10000
5000
0
-200 200 600 1000 1400 1800

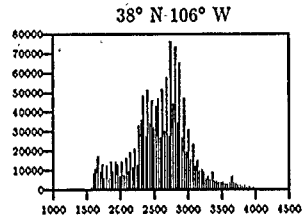

38° N 128° E
40000
35000
30000
25000
20000
15000
10000
5000
0
10 100 1000 10000

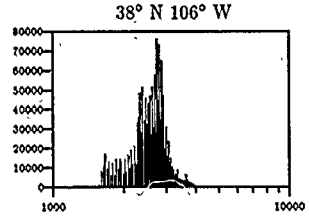

38° N 104° W
60000
50000
40000
30000
20000
10000
0
1100 1200 1300 1400 1500 1600 1700 1800 1900

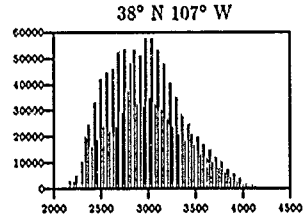

38° N 104° W
60000
50000
40000
30000
20000
10000
0
1000 10000

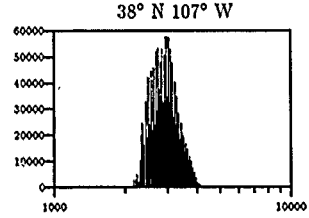

38° N 105° W
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
1000 1500 2000 2500 3000 3500 4000

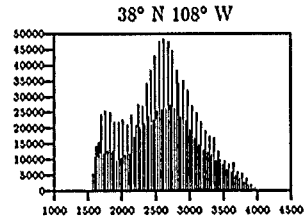

38° N 105° W
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
1000
10000

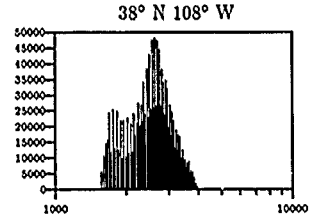

38° N 106° W
80000
70000
60000
50000
40000
30000
20000
10000
0
1000 1500 2000 2500 3000 3500 4000 4500

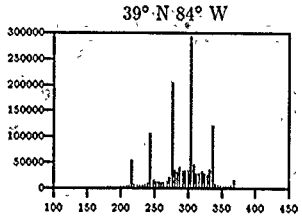

38° N 106° W
80000
70000
60000
50000
40000
30000
20000
10000
0
1000
10000

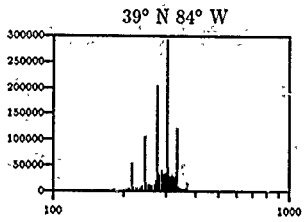

38° N 107° W
60000
50000
40000
30000
20000
10000
0
2000 2500 3000 3500 4000 4500

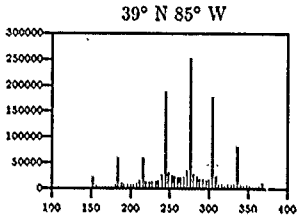

38° N 107° W
60000
50000
40000
30000
20000
10000
0
1000
10000

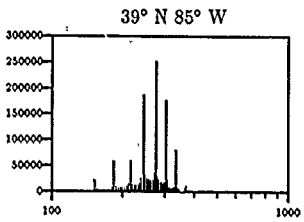

38° N 108° W
50000
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
1000 1500 2000 2500 3000 3500 4000 4500

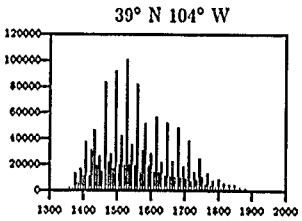

38° N 108° W
50000
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
1000
10000

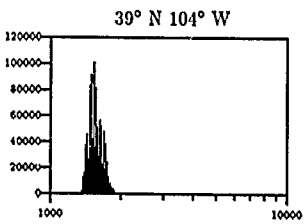

39° N 84° W
300000
250000
200000
150000
100000
50000
0
100 150 200 250 300 350 400 450

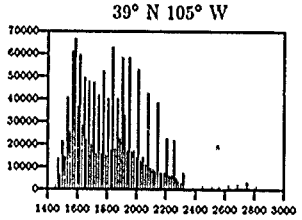

39° N 84° W
300000
250000
200000
150000
100000
50000
0
100
1000

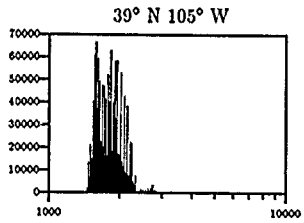

39° N 85° W
300000
250000
200000
150000
100000
50000
0
100 150 200 250 300 350 400

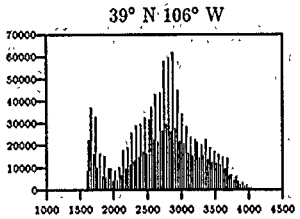

39° N 85° W
300000
250000
200000
150000
100000
50000
0
100
1000

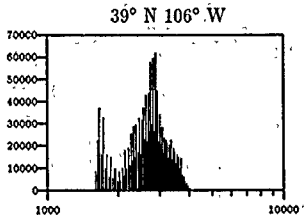

39° N 104° W
120000
100000
80000
60000
40000
20000
0
1300 1400 1500 1600 1700 1800 1900 2000

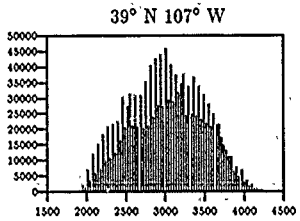

39° N 104° W
120000
100000
80000
60000
40000
20000
0
1000
10000

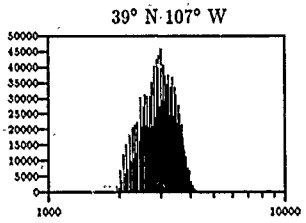

39° N 105° W
70000
60000
50000
40000
30000
20000
10000
0
1400 1600 1800 2000 2200 2400 2600 2800 3000

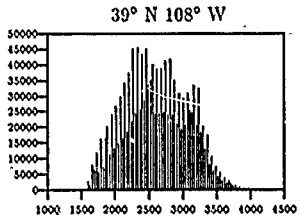

39° N 105° W
70000
60000
50000
40000
30000
20000
10000
0
1000
10000

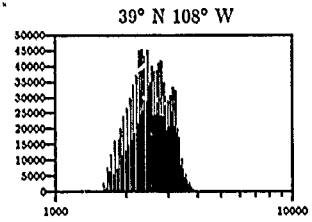

39° N 106° W
70000
60000
50000
40000
30000
20000
10000
0
1000 1500 2000 2500 3000 3500 4000 4500

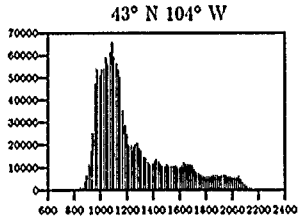

39° N 106° W
70000
60000
50000
40000
30000
20000
10000
0
1000
10000

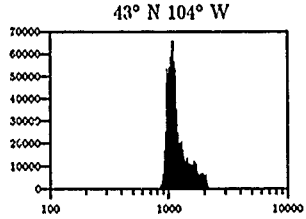

39° N 107° W
50000
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
1500 2000 2500 3000 3500 4000 4500

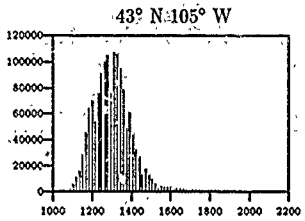

39° N 107° W
50000
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
1000
10000

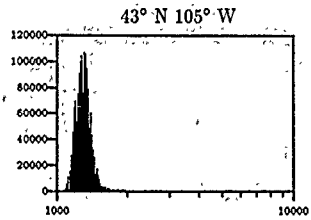

39° N 108° W
50000
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
1000 1500 2000 2500 3000 3500 4000 4500

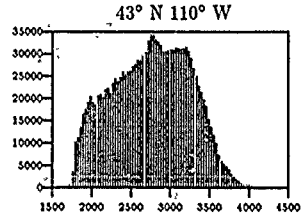

39° N 108° W
50000
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
1000
10000

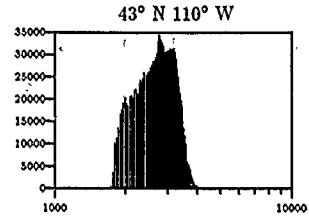

43° N 104° W
70000
60000
50000
40000
30000
20000
10000
0
600 800 1000 1200 1400 1600 1800 2000 2200 2400

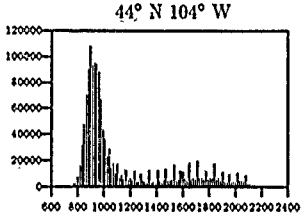

43° N 104° W
70000
60000
50000
40000
30000
20000
10000
0
100
1000
10000

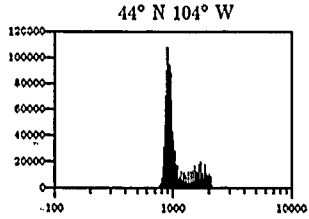

43° N 105° W
120000
100000
80000
60000
40000
20000
0
1000 1200 1400 1600 1800 2000 2200

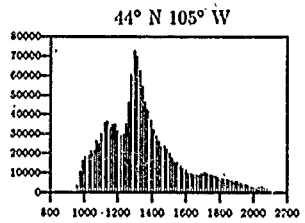

43° N 105° W
120000
100000
80000
60000
40000
20000
0
1000
10000

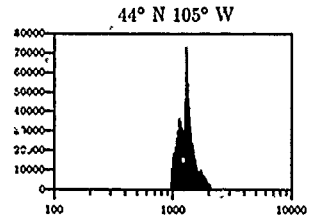

43° N 110° W
35000
30000
25000
20000
15000
10000
5000
0
1500 2000 2500 3000 3500 4000 4500

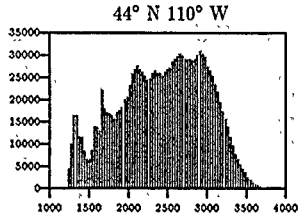

43° N 110° W
35000
30000
25000
20000
15000
10000
5000
0
1000
10000

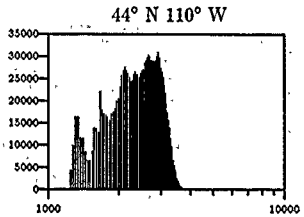

44° N 104° W
120000
100000
80000
60000
40000
20000
0
600 800 1000 1200 1400 1600 1800 2000 2200 2400

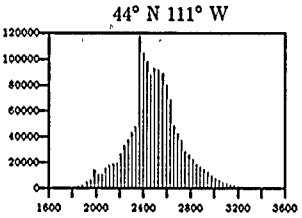

44° N 104° W
120000
100000
80000
60000
40000
20000
0
100
1000
10000

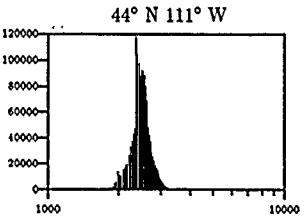

44° N 105° W
80000
70000
60000
50000
40000
30000
20000
10000
0
800 1000 1200 1400 1600 1800 2000 2200


44° N 105° W
80000
70000
60000
50000
40000
30000
20000
10000
0
100
1000
10000


44° N 110° W
35000
30000
25000
20000
15000
10000
5000
0
1000 1500 2000 2500 3000 3500 4000

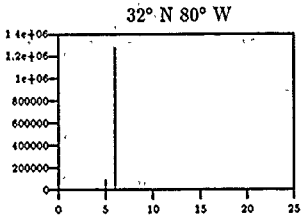

44° N 110° W
35000
30000
25000
20000
15000
10000
5000
0
1000
10000

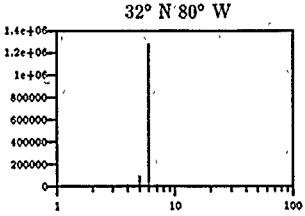

44° N 111° W
120000
100000
80000
60000
40000
20000
0
1600 2000 2400 2800 3200 3600

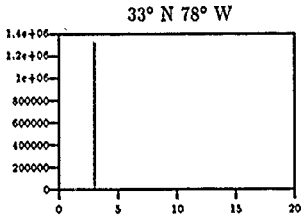

44° N 111° W
120000
100000
80000
60000
40000
20000
0
1000
10000

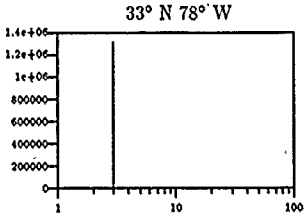
B.2 Distributions of Residual Terrain Elevation Data

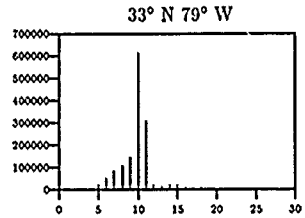
The following pages show terrain elevation histograms from full-resolution DMA DTED as taken from the DTED Evaluation Cells listed in Chapter III. Global trend has been removed from these cells with `resid` before the histograms were generated. The latitude and longitude of the southwest corner of the DTED cell used to build each histogram is provided above the histogram. All histograms are shown on both normal and log-normal scales. Zero-elevation data has been retained in these histograms as it is impossible to such data isolate after trend has been removed. The data has been shifted to be entirely above zero in order to make the log-normal plots meaningful.

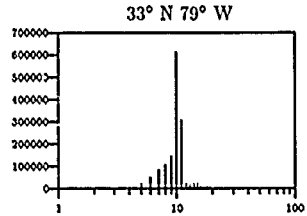

32° N 80° W
1.4e+06
1.2e+06
1e+06
800000
600000
400000
200000
0
0 5 10 15 20 25

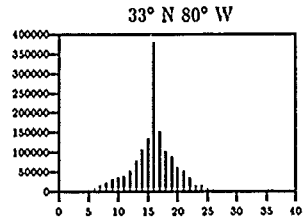

32° N 80° W
1.4e+06
1.2e+06
1e+06
800000
600000
400000
200000
0
1 10 100

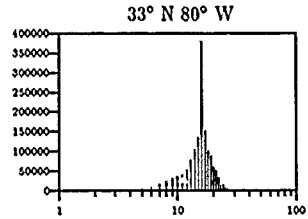

33° N 78° W
1.4e+06
1.2e+06
1e+06
800000
600000
400000
200000
0
0 5 10 15 20

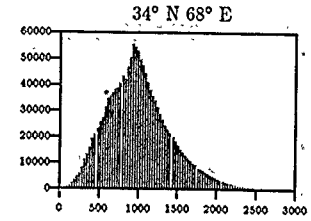

33° N 78° W
1.4e+06
1.2e+06
1e+06
800000
600000
400000
200000
0
1 10 100

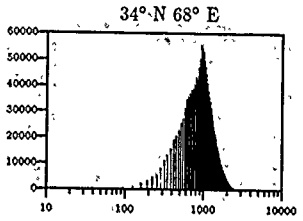

33° N 79° W
700000
600000
500000
400000
300000
200000
100000
0
0 5 10 15 20 25 30

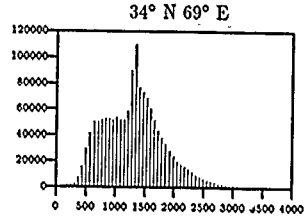

33° N 79° W
700000
600000
500000
400000
300000
200000
100000
0
1 10 100

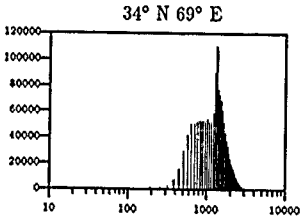

33° N 80° W
400000
350000
300000
250000
200000
150000
100000
50000
0
0 5 10 15 20 25 30 35 40

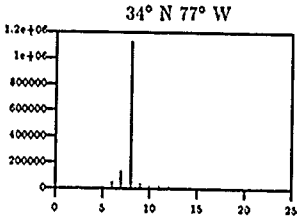

33° N 80° W
400000
350000
300000
250000
200000
150000
100000
50000
0
1 10 100

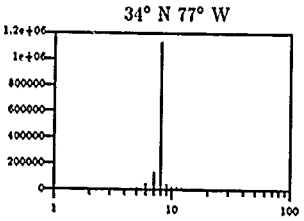

34° N 68° E
60000
50000
40000
30000
20000
10000
0
0 500 1000 1500 2000 2500 3000

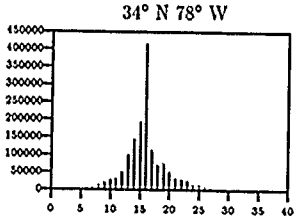

34° N 68° E
60000
50000
40000
30000
20000
10000
0
10 100 1000 10000

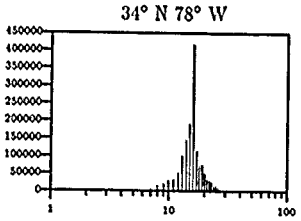

34° N 69° E
120000
100000
80000
60000
40000
20000
0
0 500 1000 1500 2000 2500 3000 3500 4000

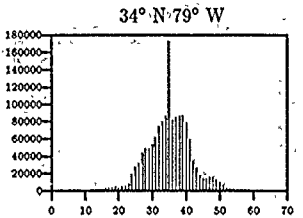

34° N 69° E
120000
100000
80000
60000
40000
20000
0
10 100 1000 10000

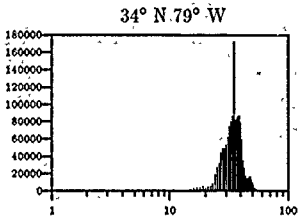

34° N 77° W
1.2e+06
1e+06
800000
600000
400000
200000
0
0 5 10 15 20 25

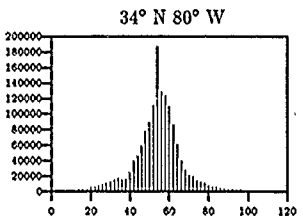

34° N 77° W
1.2e+06
1e+06
800000
600000
400000
200000
0
1 10 100

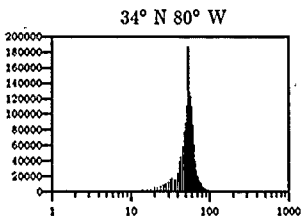

34° N 78° W
450000
400000
350000
300000
250000
200000
150000
100000
50000
0
0 5 10 15 20 25 30 35 40

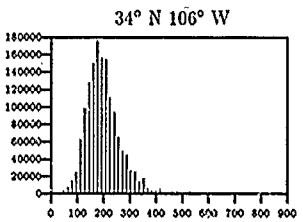

34° N 78° W
450000
400000
350000
300000
250000
200000
150000
100000
50000
0
1 10 100

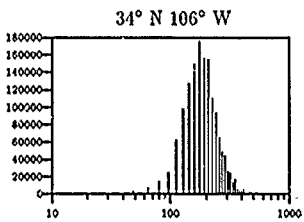

34° N 79° W
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60 70

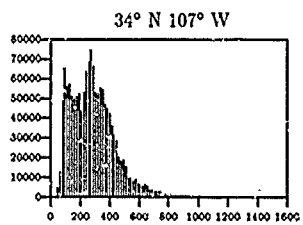

34° N 79° W
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
1 10 100

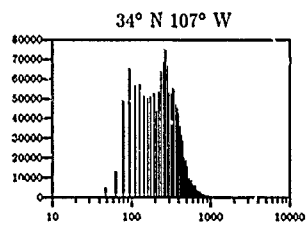

34° N 80° W
200000
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
0 20 40 60 80 100 120

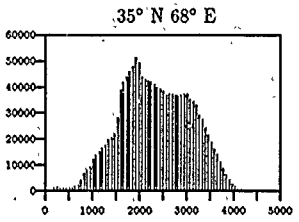

34° N 80° W
200000
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
1 10 100 1000

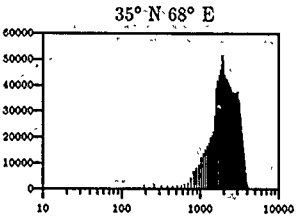

34° N 106° W
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
0 100 200 300 400 500 600 700 800 900

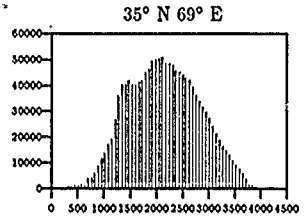

34° N 106° W
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
10 100 1000

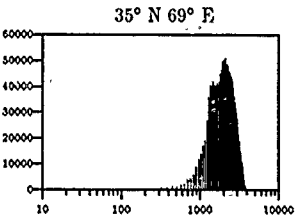

34° N 107° W
80000
70000
60000
50000
40000
30000
20000
10000
0
0 200 400 600 800 1000 1200 1400 1600

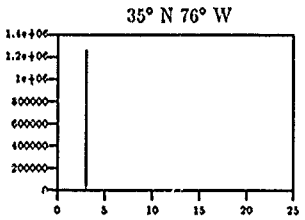

34° N 107° W
80000
70000
60000
50000
40000
30000
20000
10000
0
10 100 1000 10000

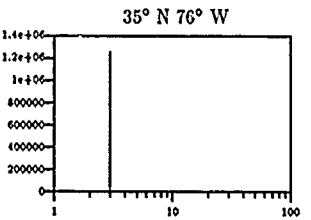

35° N 68° E
60000
50000
40000
30000
20000
10000
0
0
1000
2000
3000
4000
5000

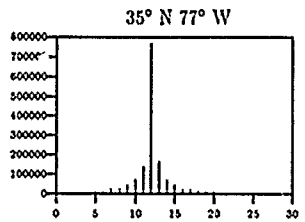

35° N 68° E
60000
50000
40000
30000
20000
10000
0
10
100
1000
10000

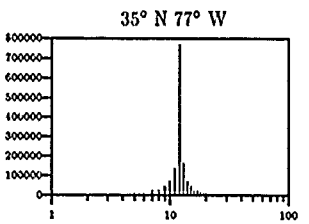

35° N 69° E
60000
50000
40000
30000
20000
10000
0
0
500
1000
1500
2000
2500
3000
3500
4000
4500

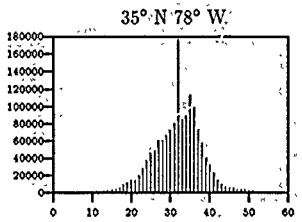

35° N 69° E
60000
50000
40000
30000
20000
10000
0
10
100
1000
10000

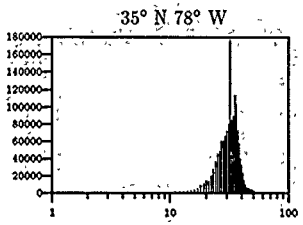

35° N 76° W
1.4e+06
1.2e+06
1e+06
800000
600000
400000
200000
0
0
5
10
15
20
25

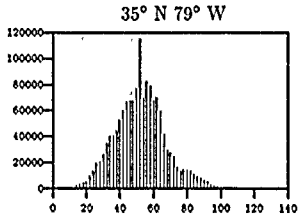

35° N 76° W
1.4e+06
1.2e+06
1e+06
800000
600000
400000
200000
0
1
10
100

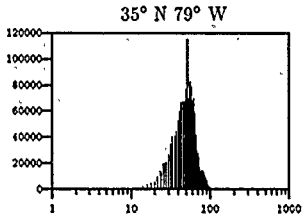

35° N 77° W
800000
700000
600000
500000
400000
300000
200000
100000
0
0
5
10
15
20
25
30

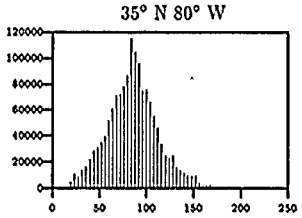

35° N 77° W
800000
700000
600000
500000
400000
300000
200000
100000
0
1
10
100

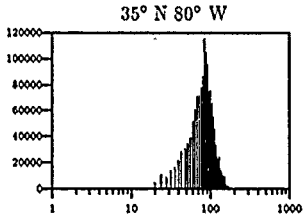

35° N 78° W
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

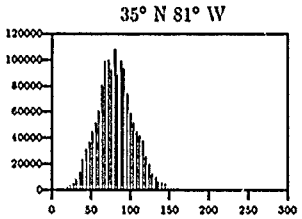

35° N 78° W
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
1 10 100

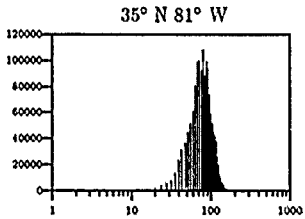

35° N 79° W
120000
100000
80000
60000
40000
20000
0
0 20 40 60 80 100 120 140

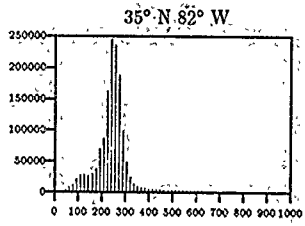

35° N 79° W
120000
100000
80000
60000
40000
20000
0
1 10 100 1000

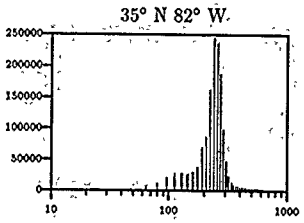

35° N 80° W
120000
100000
80000
60000
40000
20000
0
0 50 100 150 200 250

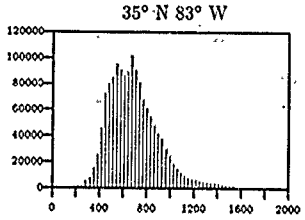

35° N 80° W
120000
100000
80000
60000
40000
20000
0
1 10 100 1000

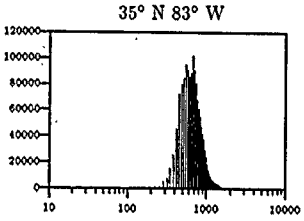

35° N 81° W
120000
100000
80000
60000
40000
20000
0
0 50 100 150 200 250 300

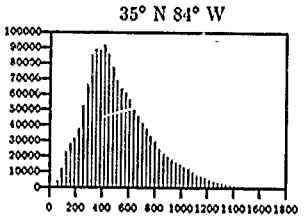

35° N 81° W
120000
100000
80000
60000
40000
20000
0
1 10 100 1000

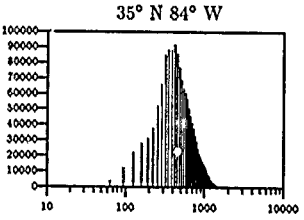

35° N 82° W
250000
200000
150000
100000
50000
0
0 100 200 300 400 500 600 700 800 900 1000

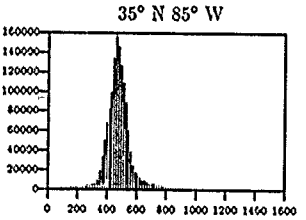

35° N 82° W
250000
200000
150000
100000
50000
0
10
100
1000

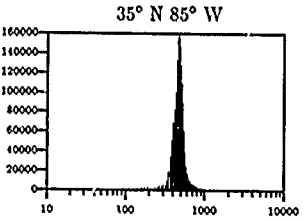

35° N 83° W
120000
100000
80000
60000
40000
20000
0
0 400 800 1200 1600 2000


35° N 83° W
120000
100000
80000
60000
40000
20000
0
10
100
1000
10000

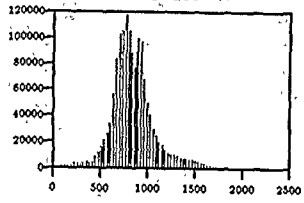

35° N 84° W
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
0 200 400 600 800 1000 1200 1400 1600 1800


35° N 84° W
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
10
100
1000
10000

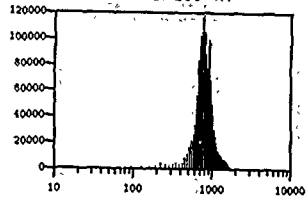

35° N 85° W
160000
140000
120000
100000
80000
60000
40000
20000
0
0 200 400 600 800 1000 1200 1400 1600


35° N 85° W
160000
140000
120000
100000
80000
60000
40000
20000
0
10
100
1000
10000

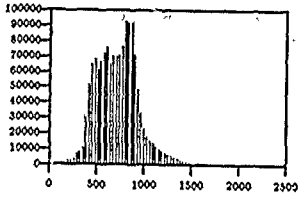
35° N 106° W



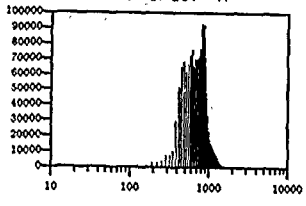
35° N 106° W



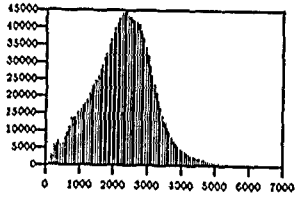
35° N 107° W



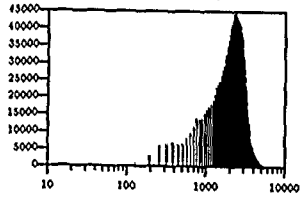
35° N 107° W



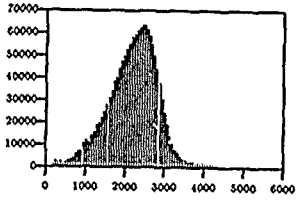
36° N 74° E



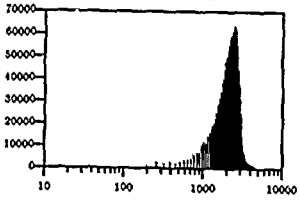
36° N 74° E

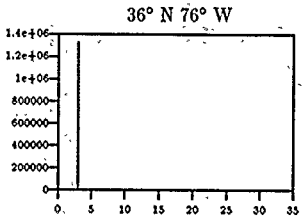


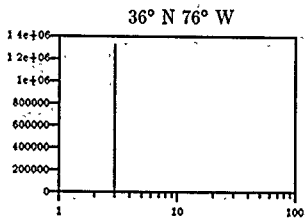
36° N 75° E

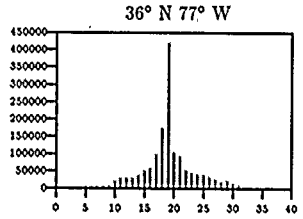


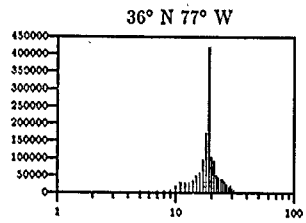
36° N 75° E

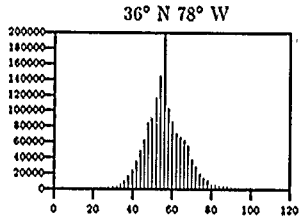


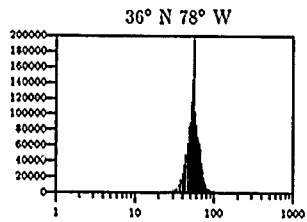

36° N 76° W
1.4e+06
1.2e+06
1e+06
800000
600000
400000
200000
0
0 5 10 15 20 25 30 35

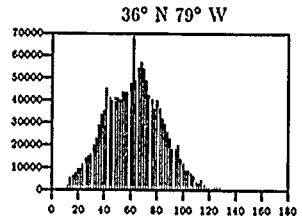

36° N 76° W
1.4e+06
1.2e+06
1e+06
800000
600000
400000
200000
0
1 10 100

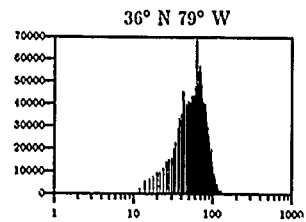

36° N 77° W
450000
400000
350000
300000
250000
200000
150000
100000
50000
0
0 5 10 15 20 25 30 35 40

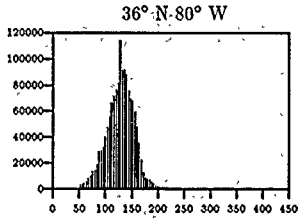

36° N 77° W
450000
400000
350000
300000
250000
200000
150000
100000
50000
0
1 10 100

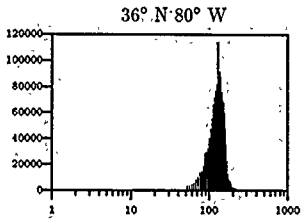

36° N 78° W
200000
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
0 20 40 60 80 100 120

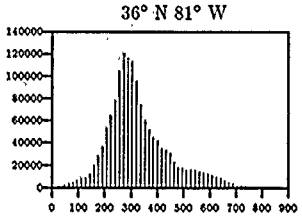

36° N 78° W
200000
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
1 10 100 1000

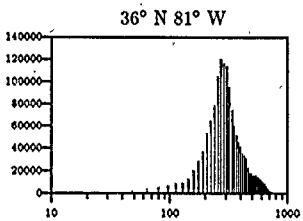

36° N 79° W
70000
60000
50000
40000
30000
20000
10000
0
0 20 40 60 80 100 120 140 160 180

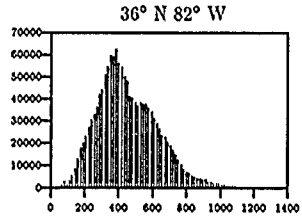

36° N 79° W
70000
60000
50000
40000
30000
20000
10000
0
1 10 100 1000

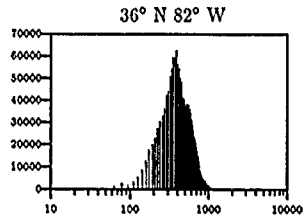

36° N 80° W
120000
100000
80000
60000
40000
20000
0
0 50 100 150 200 250 300 350 400 450

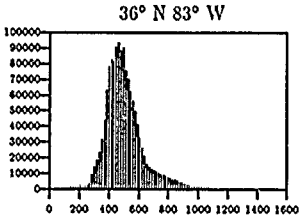

36° N 80° W
120000
100000
80000
60000
40000
20000
0
1 10 100 1000

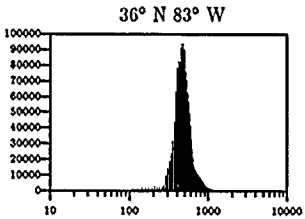

36° N 81° W
140000
120000
100000
80000
60000
40000
20000
0
0 100 200 300 400 500 600 700 800 900

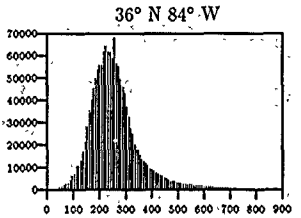

36° N 81° W
140000
120000
100000
80000
60000
40000
20000
0
10 100 1000

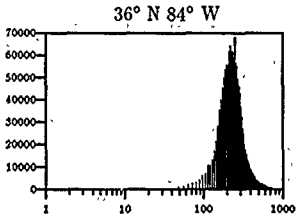

36° N 82° W
70000
60000
50000
40000
30000
20000
10000
0
0 200 400 600 800 1000 1200 1400

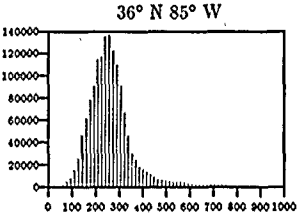

36° N 82° W
70000
60000
50000
40000
30000
20000
10000
0
10 100 1000 10000

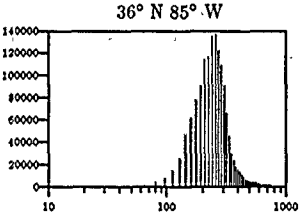

36° N 83° W
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
0 200 400 600 800 1000 1200 1400 1600

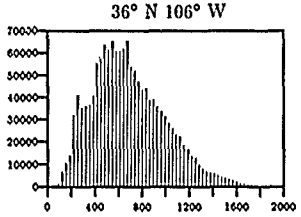

36° N 83° W
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
10 100 1000 10000

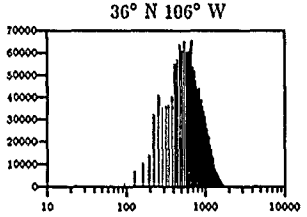

36° N 84° W
70000
60000
50000
40000
30000
20000
10000
0
0 100 200 300 400 500 600 700 800 900

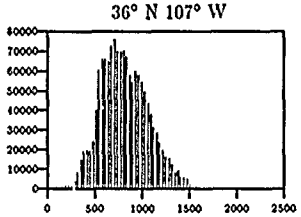

36° N 84° W
70000
60000
50000
40000
30000
20000
10000
0
1 10 100 1000

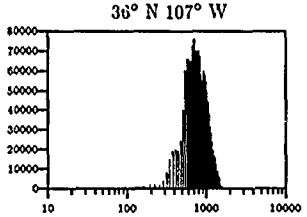

36° N 85° W
140000
120000
100000
80000
60000
40000
20000
0
0 100 200 300 400 500 600 700 800 900 1000

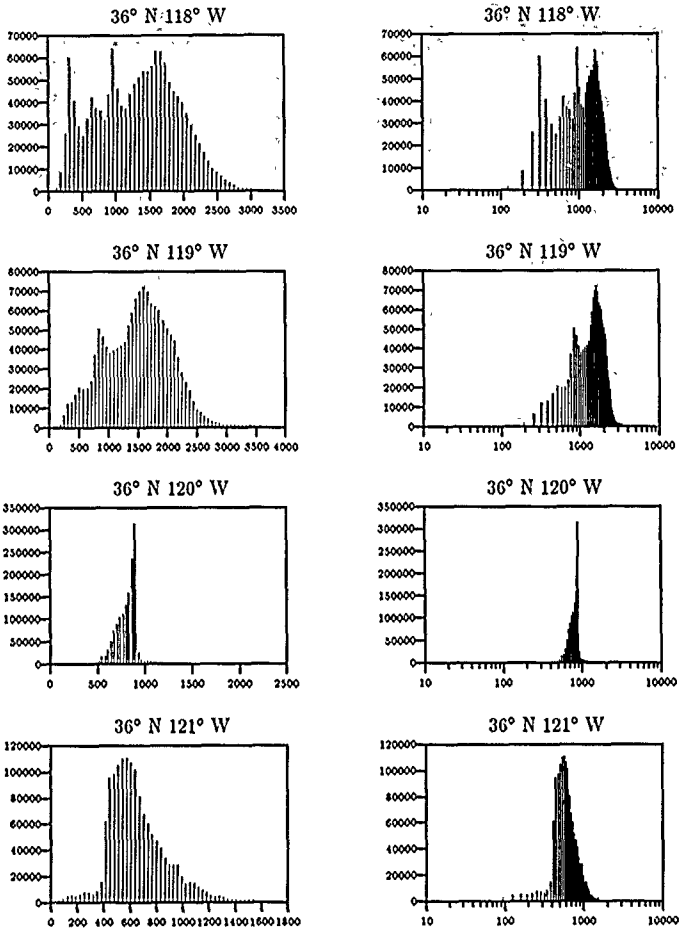

36° N 85° W
140000
120000
100000
80000
60000
40000
20000
0
10 100 1000

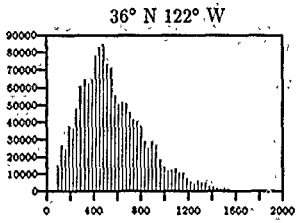

36° N 106° W
70000
60000
50000
40000
30000
20000
10000
0
0 400 800 1200 1600 2000

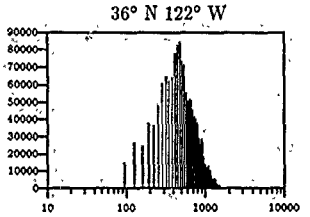

36° N 106° W
70000
60000
50000
40000
30000
20000
10000
0
10 100 1000 10000

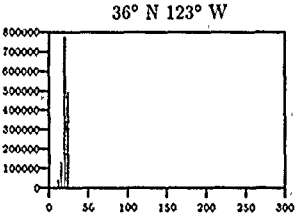

36° N 107° W
80000
70000
60000
50000
40000
30000
20000
10000
0
0 500 1000 1500 2000 2500

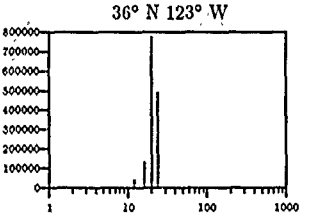

36° N 107° W
80000
70000
60000
50000
40000
30000
20000
10000
0
10 100 1000 10000

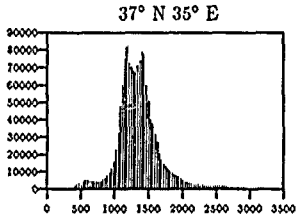

36° N 118° W
36° N 118° W
36° N 119° W
36° N 119° W
36° N 120° W
36° N 120° W
36° N 121° W
36° N 121° W

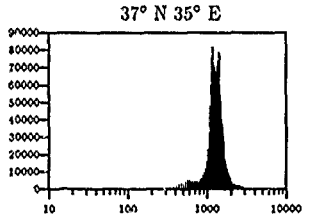

36° N 122° W
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
0
400
800
1200
1600
2000

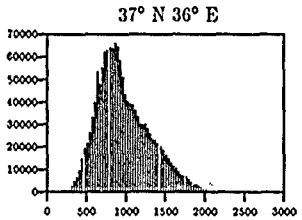

36° N 122° W
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
10
100
1000
10000

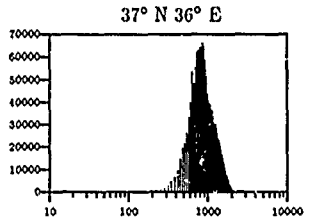

36° N 123° W
800000
700000
600000
500000
400000
300000
200000
100000
0
0
50
100
150
200
250
300

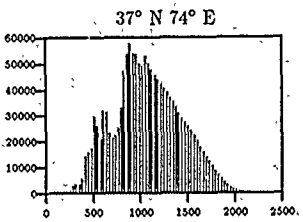

36° N 123° W
800000
700000
600000
500000
400000
300000
200000
100000
0
1
10
100
1000

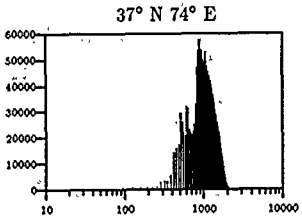

37° N 35° E
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
0
500
1000
1500
2000
2500
3000
3500

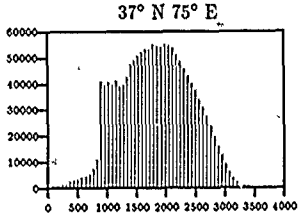

37° N 35° E
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
10
100
1000
10000

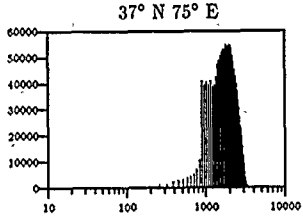

37° N 36° E
70000
60000
50000
40000
30000
20000
10000
0
0
500
1000
1500
2000
2500
3000

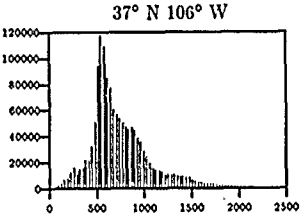

37° N 36° E
70000
60000
50000
40000
30000
20000
10000
0
10
100
1000
10000

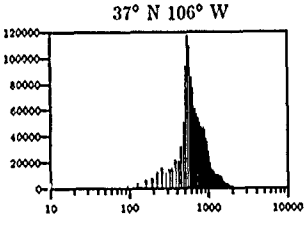

37° N 74° E
60000
50000
40000
30000
20000
10000
0
0 500 1000 1500 2000 2500

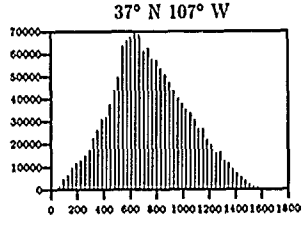

37° N 74° E
60000
50000
40000
30000
20000
10000
0
10 100 1000 10000

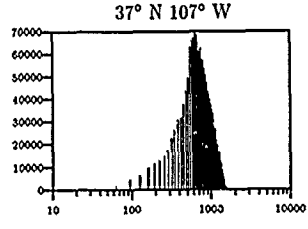

37° N 75° E
60000
50000
40000
30000
20000
10000
0
0 500 1000 1500 2000 2500 3000 3500 4000

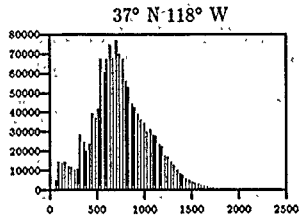

37° N 75° E
60000
50000
40000
30000
20000
10000
0
10 100 1000 10000

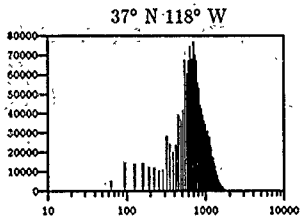

37° N 106° W
120000
100000
80000
60000
40000
20000
0
0 500 1000 1500 2000 2500

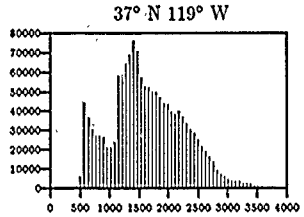

37° N 106° W
120000
100000
80000
60000
40000
20000
0
10 100 1000 10000

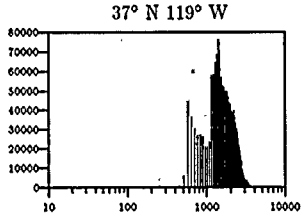

37° N 107° W
70000
60000
50000
40000
30000
20000
10000
0
0 200 400 600 800 1000 1200 1400 1600 1800

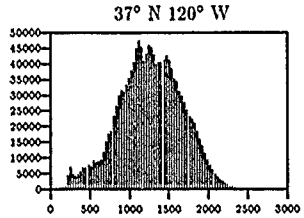

37° N 107° W
70000
60000
50000
40000
30000
20000
10000
0
10 100 1000 10000

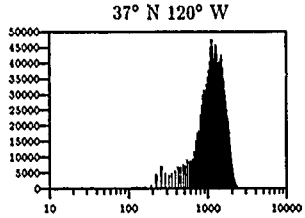

37° N 118° W
80000
70000
60000
50000
40000
30000
20000
10000
0
0 500 1000 1500 2000 2500

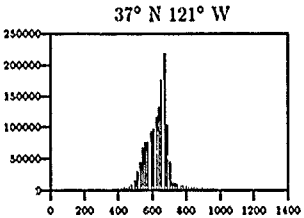

37° N 118° W
80000
70000
60000
50000
40000
30000
20000
10000
0
10 100 1000 10000

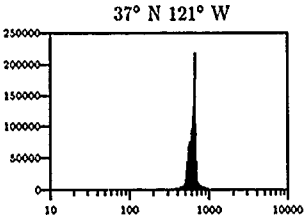

37° N 119° W
80000
70000
60000
50000
40000
30000
20000
10000
0
0 500 1000 1500 2000 2500 3000 3500 4000

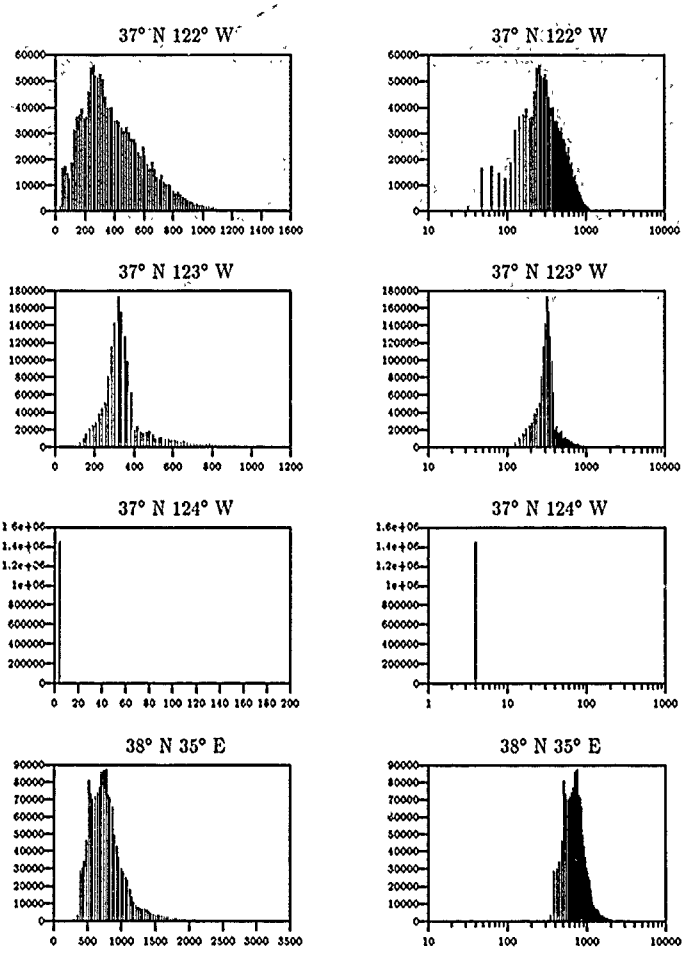

37° N 119° W
80000
70000
60000
50000
40000
30000
20000
10000
0
10 100 1000 10000

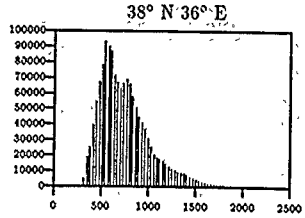

37° N 120° W
50000
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
0 500 1000 1500 2000 2500 3000

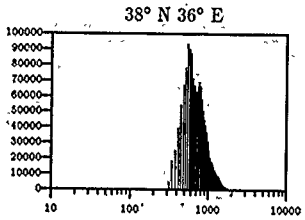

37° N 120° W
50000
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
10 100 1000 10000

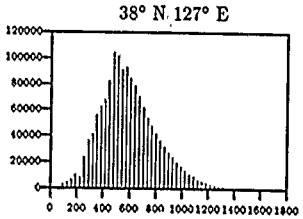

37° N 121° W
250000
200000
150000
100000
50000
0
0 200 400 600 800 1000 1200 1400

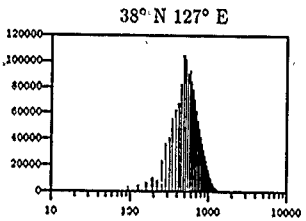

37° N 121° W
250000
200000
150000
100000
50000
0
10 100 1000 10000

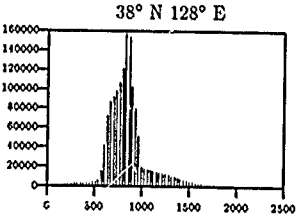

37° N 122° W
37° N 122° W
37° N 123° W
37° N 123° W
37° N 124° W
37° N 124° W
38° N 35° E
38° N 35° E

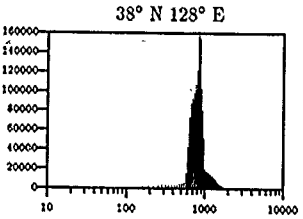

38° N 36° E
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
0
500
1000
1500
2000
2500

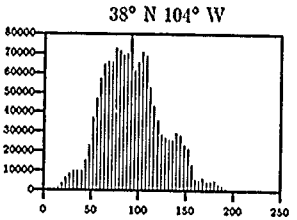

38° N 36° E
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
10
100
1000
10000

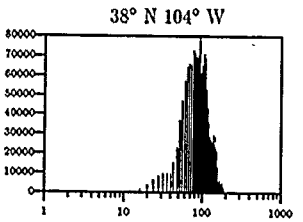

38° N 127° E
120000
100000
80000
60000
40000
20000
0
0
200
400
600
800
1000
1200
1400
1600
1800

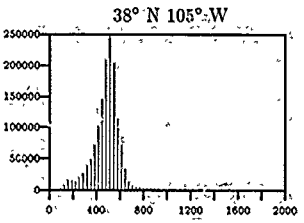

38° N 127° E
120000
100000
80000
60000
40000
20000
0
10
100
1000
10000

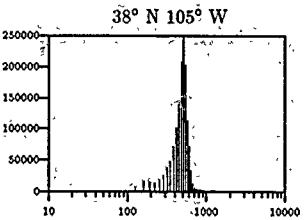

38° N 128° E
160000
140000
120000
100000
80000
60000
40000
20000
0
0
500
1000
1500
2000
2500

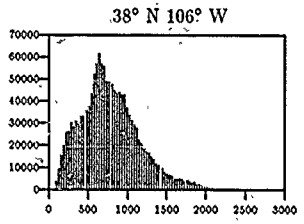

38° N 128° E
160000
140000
120000
100000
80000
60000
40000
20000
0
10
100
1000
10000

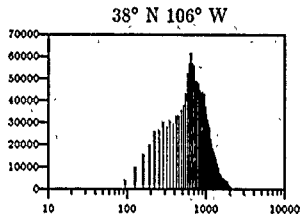

38° N 104° W
80000
70000
60000
50000
40000
30000
20000
10000
0
0
50
100
150
200
250

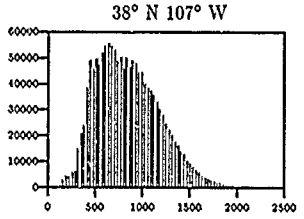

38° N 104° W
80000
70000
60000
50000
40000
30000
20000
10000
0
1
10
100
1000

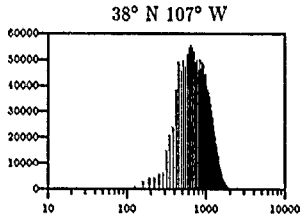

38° N 105° W
250000
200000
150000
100000
50000
0
0
400
800
1200
1600
2000

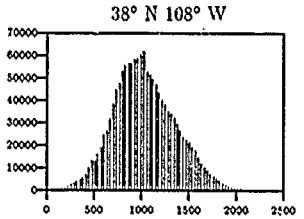

38° N 105° W
250000
200000
150000
100000
50000
0
10
100
1000
10000

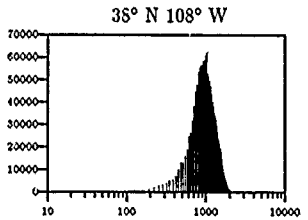

38° N 106° W
70000
60000
50000
40000
30000
20000
10000
0
0
500
1000
1500
2000
2500
3000


38° N 106° W
70000
60000
50000
40000
30000
20000
10000
0
10
100
1000
10000

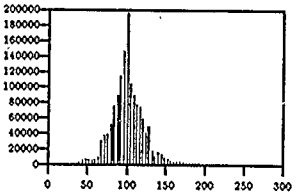

38° N 107° W
60000
50000
40000
30000
20000
10000
0
0
500
1000
1500
2000
2500


38° N 107° W
60000
50000
40000
30000
20000
10000
0
10
100
1000
10000

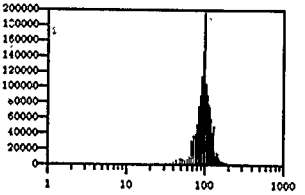

38° N 108° W
70000
60000
50000
40000
30000
20000
10000
0
0
500
1000
1500
2000
2500


38° N 108° W
70000
60000
50000
40000
30000
20000
10000
0
10
100
1000
10000

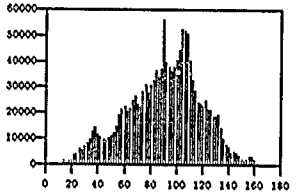
39° N 84° W



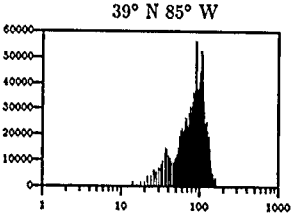
39° N 84° W



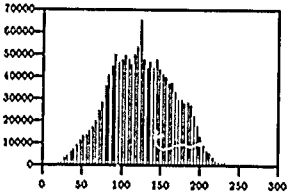
39° N 85° W



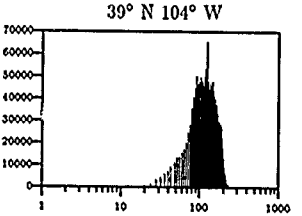
39° N 85° W



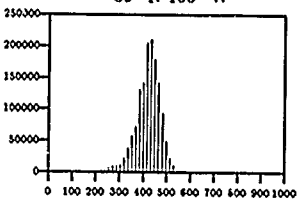
39° N 104° W



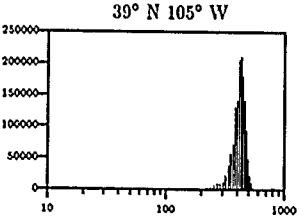
39° N 104° W

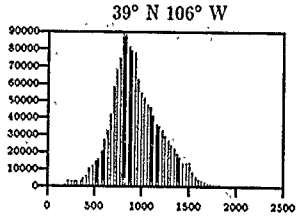


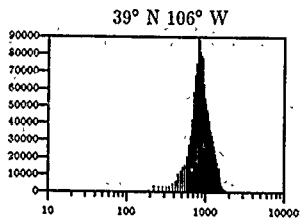
39° N 105° W

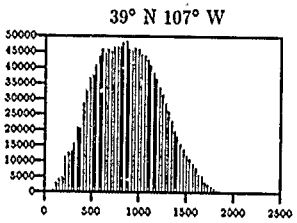


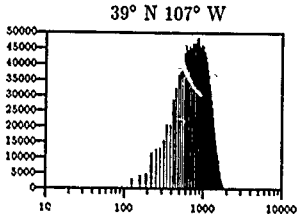
39° N 105° W

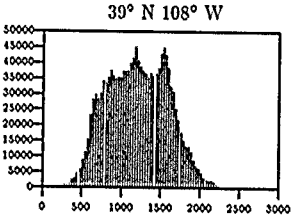


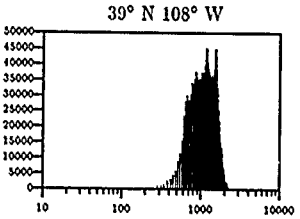

39° N 106° W
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
0 500 1000 1500 2000 2500

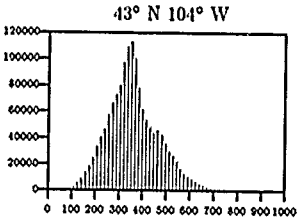

39° N 106° W
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
10 100 1000 10000

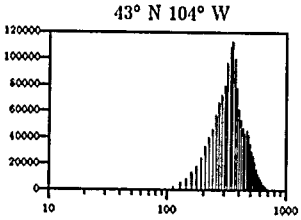

39° N 107° W
50000
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
0 500 1000 1500 2000 2500

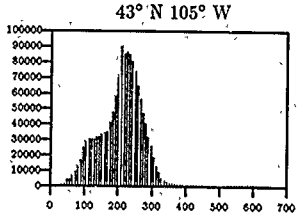

39° N 107° W
50000
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
10 100 1000 10000

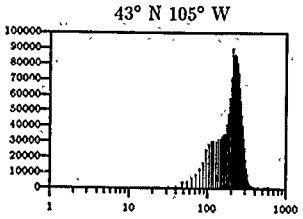

39° N 108° W
50000
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
0 500 1000 1500 2000 2500 3000

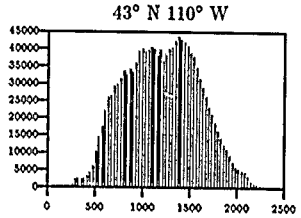

39° N 108° W
50000
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
10 100 1000 10000

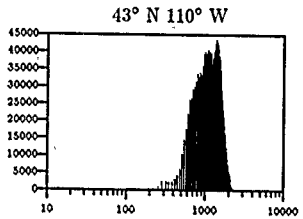

43° N 104° W
120000
100000
80000
60000
40000
20000
0
0 100 200 300 400 500 600 700 800 900 1000

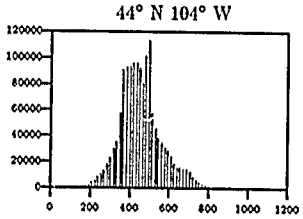

43° N 104° W
120000
100000
80000
60000
40000
20000
0
10 100 1000

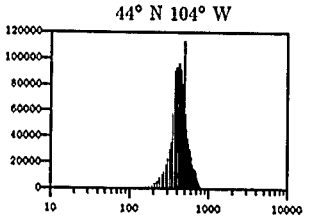

43° N 105° W
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
0 100 200 300 400 500 600 700

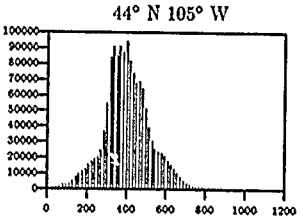

43° N 105° W
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
1 10 100 1000

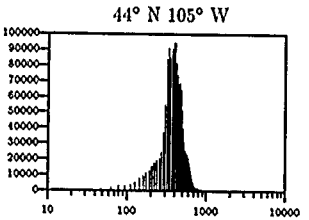

43° N 110° W
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
0 500 1000 1500 2000 2500

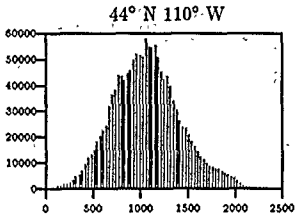

43° N 110° W
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
10 100 1000 10000

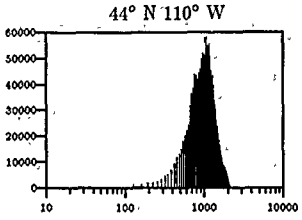

44° N 104° W
120000
100000
80000
60000
40000
20000
0
0 200 400 600 800 1000 1200

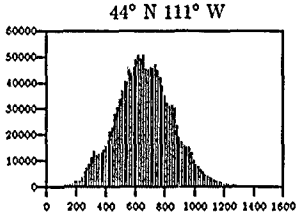

44° N 104° W
120000
100000
80000
60000
40000
20000
0
10 100 1000 10000

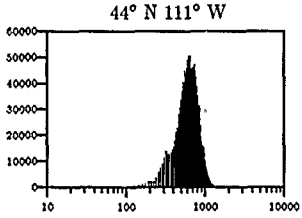

44° N 105° W
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
0 200 400 600 800 1000 1200


44° N 105° W
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
10 100 1000 10000


44° N 110° W
60000
50000
40000
30000
20000
10000
0
0
500
1000
1500
2000
2500


44° N 110° W
60000
50000
40000
30000
20000
10000
0
10
100
1000
10000

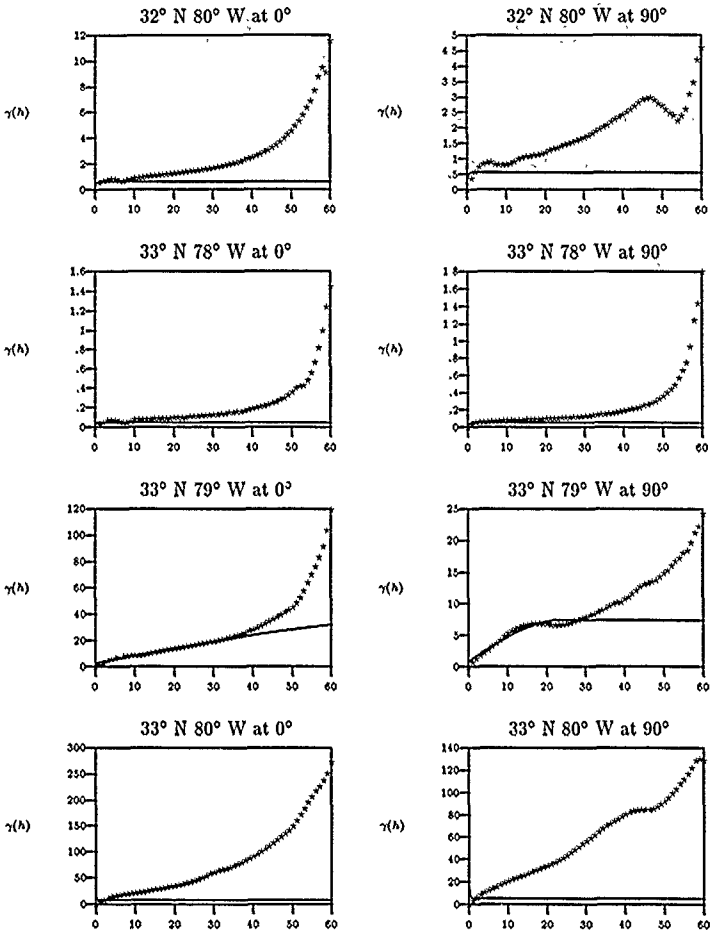

44° N 111° W
60000
50000
40000
30000
20000
10000
0
0
200
400
600
800
1000
1200
1400
1600

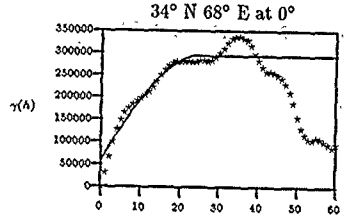

44° N 111° W
60000
50000
40000
30000
20000
10000
0
10
100
1000
10000

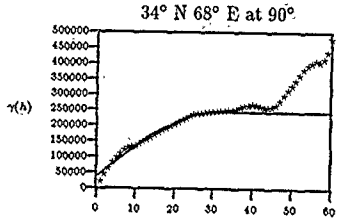
Appendix C. *Terrain Elevation Variograms*

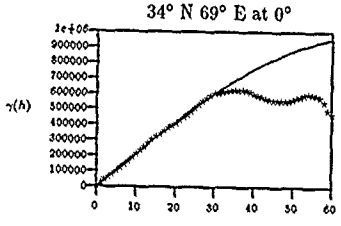
C.1 Variograms of the DTED Evaluation Cells

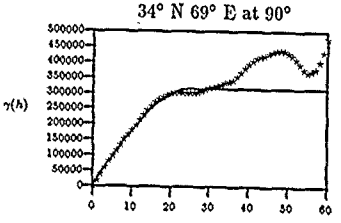
The following pages show directional experimental variograms and spherical variogram models for the DTED Evaluation Cells listed in Chapter III. These variograms were generated by `varfit` on *non-residual* terrain elevation data from these cells. The latitude and longitude of the southwest corner of the DTED cell used to generate each variogram is provided above the variogram. Also indicated is the angle at which the experimental variogram was generated (in degrees clockwise from north). The experimental variograms are denoted by the stars ($\star$), while the spherical variogram models are denoted by solid lines.

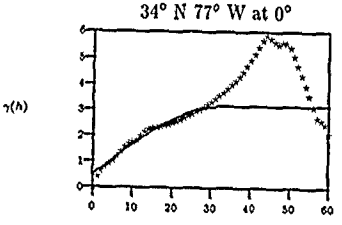

32° N 80° W at 0°
32° N 80° W at 90°
33° N 78° W at 0°
33° N 78° W at 90°
33° N 79° W at 0°
33° N 79° W at 90°
33° N 80° W at 0°
33° N 80° W at 90°
γ(h)

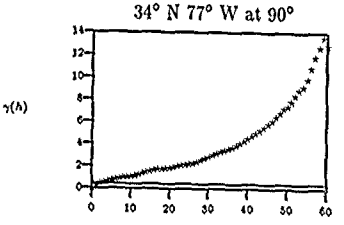

34° N 68° E at 0°
γ(h)
350000
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

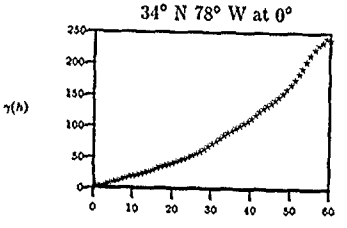

34° N 68° E at 90°
γ(h)
500000
450000
400000
350000
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

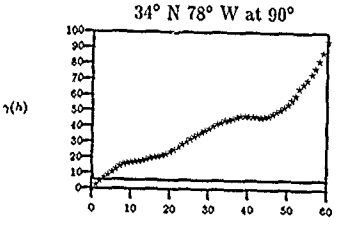

34° N 69° E at 0°
γ(h)
1e+06
900000
800000
700000
600000
500000
400000
300000
200000
100000
0
0 10 20 30 40 50 60

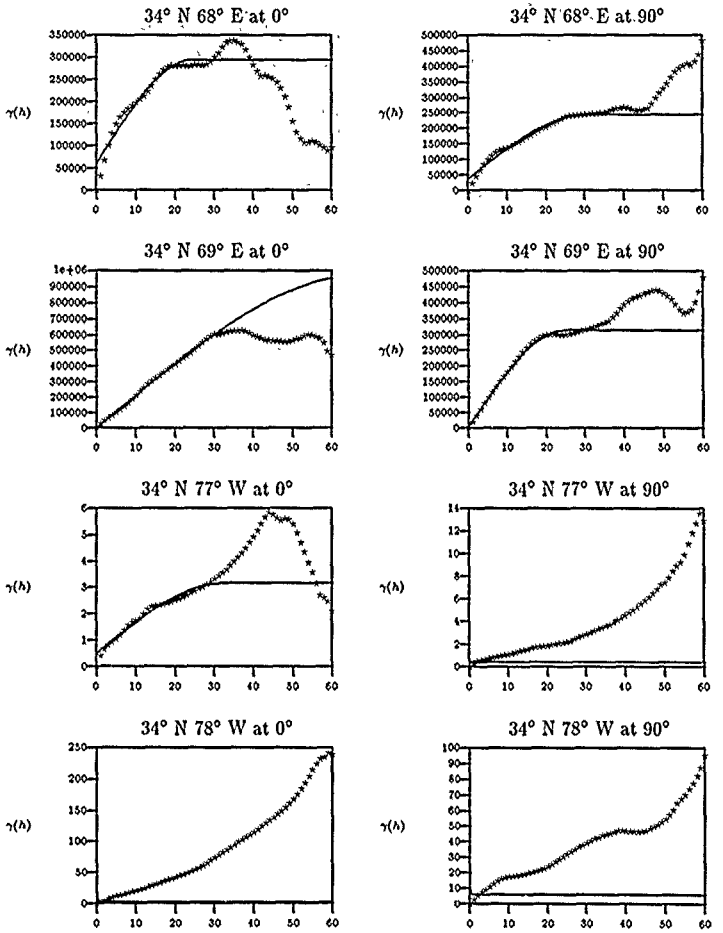

34° N 69° E at 90°
γ(h)
500000
450000
400000
350000
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

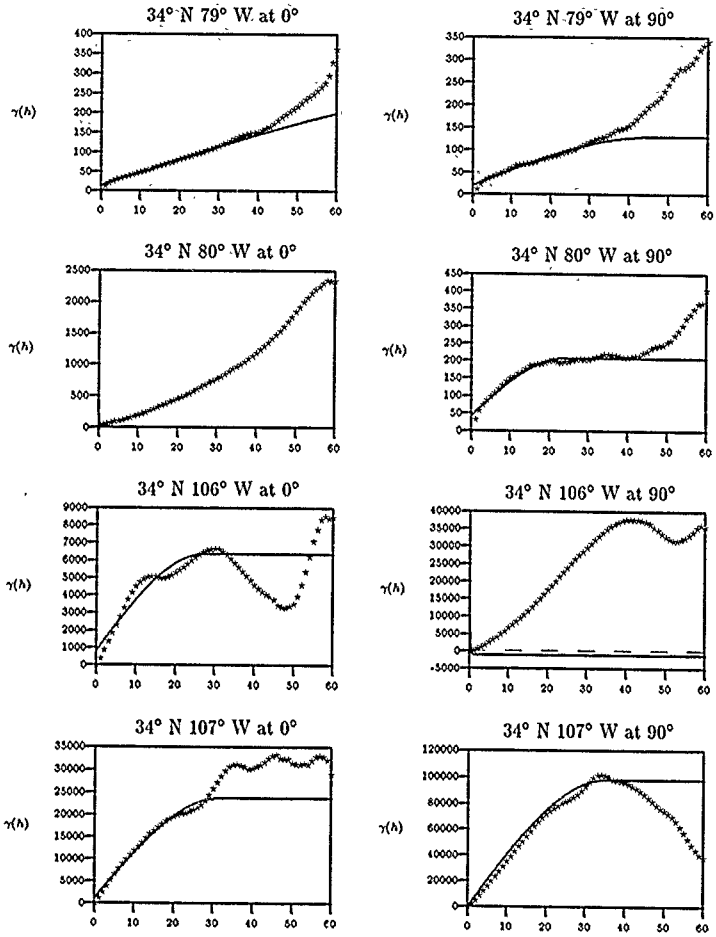

34° N 77° W at 0°
γ(h)
6
5
4
3
2
1
0
0 10 20 30 40 50 60

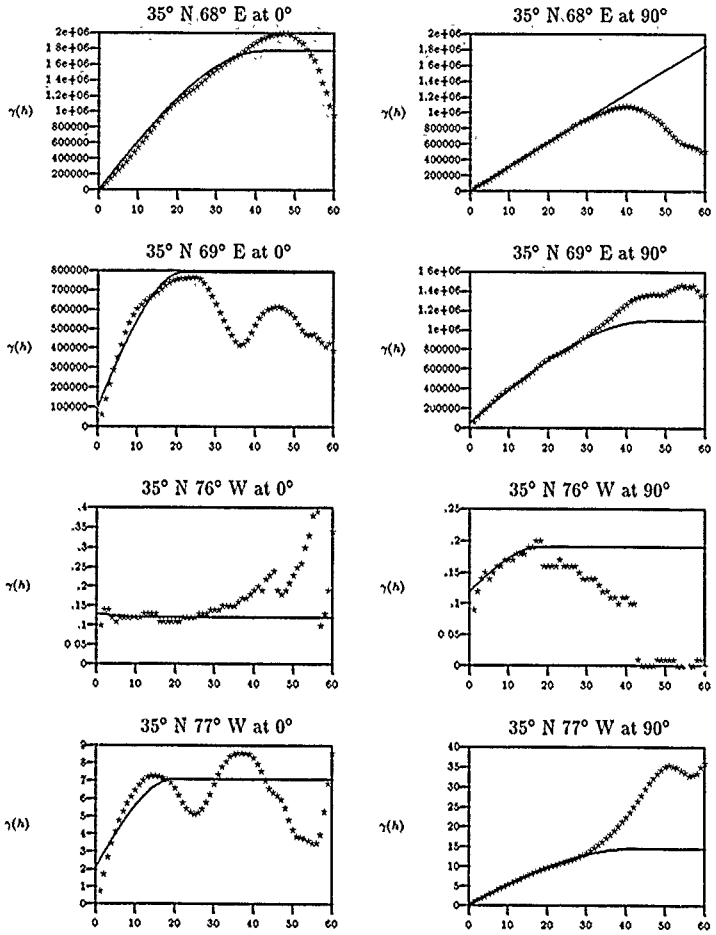

34° N 77° W at 90°
γ(h)
14
12
10
8
6
4
2
0
0 10 20 30 40 50 60

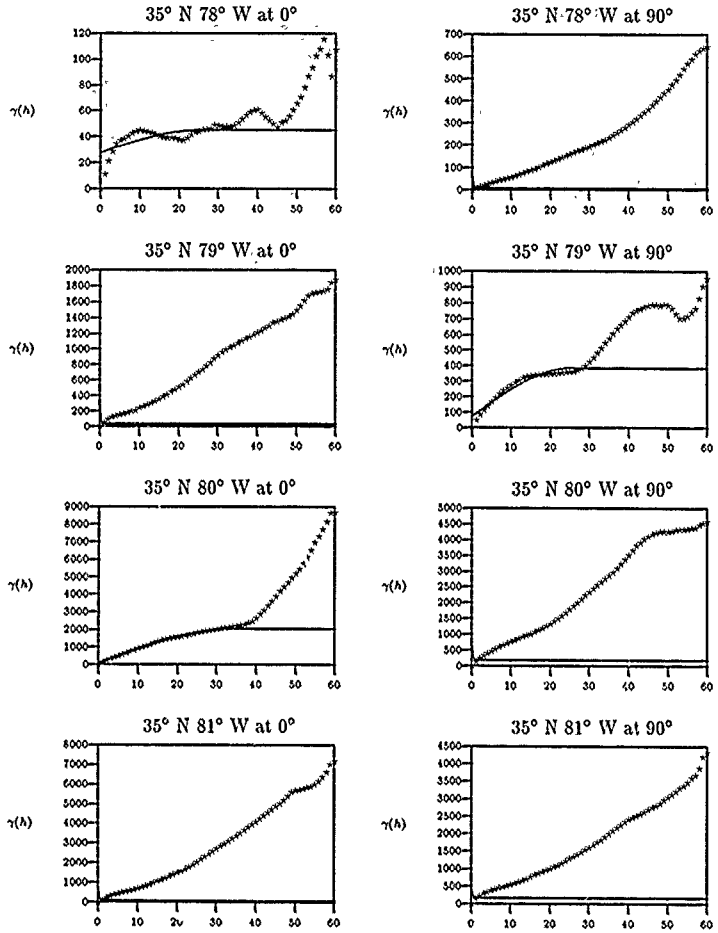

34° N 78° W at 0°
γ(h)
250
200
150
100
50
0
0 10 20 30 40 50 60

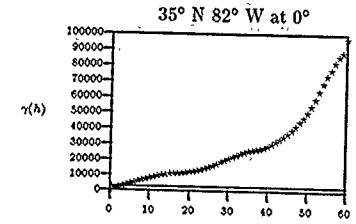

34° N 78° W at 90°
γ(h)
100
90
80
70
60
50
40
30
20
10
0
0 10 20 30 40 50 60

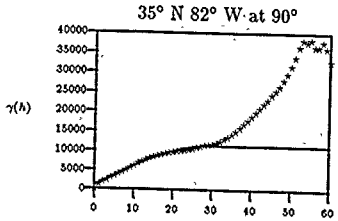

34° N 68° E at 0°
34° N 68° E at 90°
34° N 69° E at 0°
34° N 69° E at 90°
34° N 77° W at 0°
34° N 77° W at 90°
34° N 78° W at 0°
34° N 78° W at 90°
γ(h)

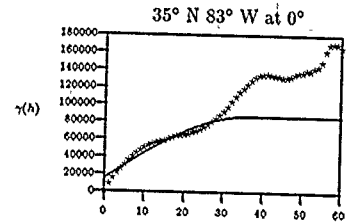

34° N 79° W at 0°
34° N 79° W at 90°
34° N 80° W at 0°
34° N 80° W at 90°
34° N 106° W at 0°
34° N 106° W at 90°
34° N 107° W at 0°
34° N 107° W at 90°
γ(h)

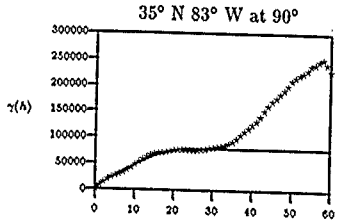

35° N 68° E at 0°
35° N 68° E at 90°
35° N 69° E at 0°
35° N 69° E at 90°
35° N 76° W at 0°
35° N 76° W at 90°
35° N 77° W at 0°
35° N 77° W at 90°
γ(h)

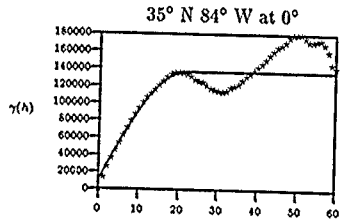

35° N 78° W at 0°
35° N 78° W at 90°
35° N 79° W at 0°
35° N 79° W at 90°
35° N 80° W at 0°
35° N 80° W at 90°
35° N 81° W at 0°
35° N 81° W at 90°
γ(h)

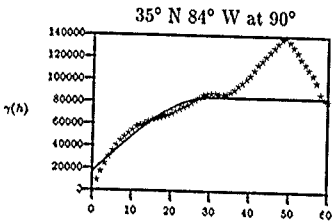

35° N 82° W at 0°
γ(h)
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
0 10 20 30 40 50 60

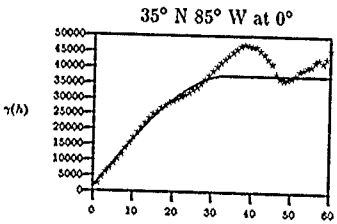

35° N 82° W at 90°
γ(h)
40000
35000
30000
25000
20000
15000
10000
5000
0
0 10 20 30 40 50 60

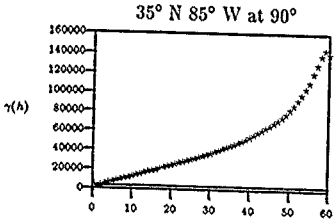

35° N 83° W at 0°
γ(h)
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

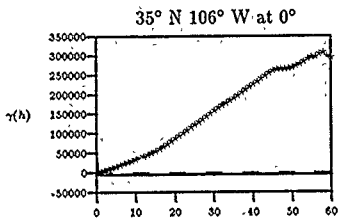

35° N 83° W at 90°
γ(h)
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

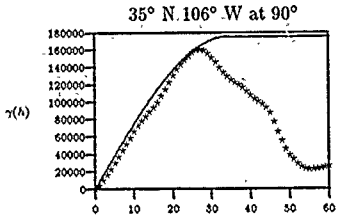

35° N 84° W at 0°
γ(h)
160000
160000
140000
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

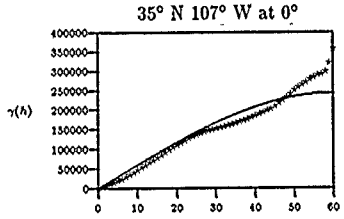

35° N 84° W at 90°
γ(h)
140000
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

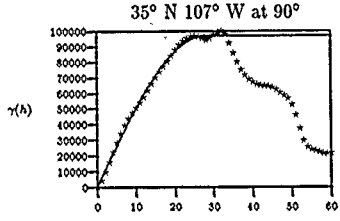

35° N 85° W at 0°
γ(h)
50000
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
0 10 20 30 40 50 60

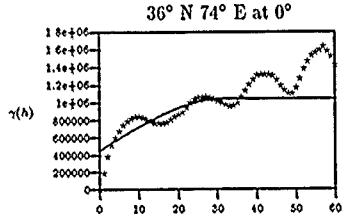

35° N 85° W at 90°
γ(h)
160000
140000
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

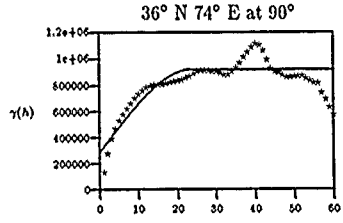

35° N 106° W at 0°
γ(h)
350000
300000
250000
200000
150000
100000
50000
0
-50000
0 10 20 30 40 50 60

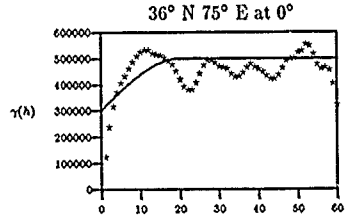

35° N 106° W at 90°
γ(h)
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

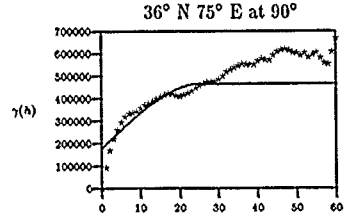

35° N 107° W at 0°
γ(h)
400000
350000
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

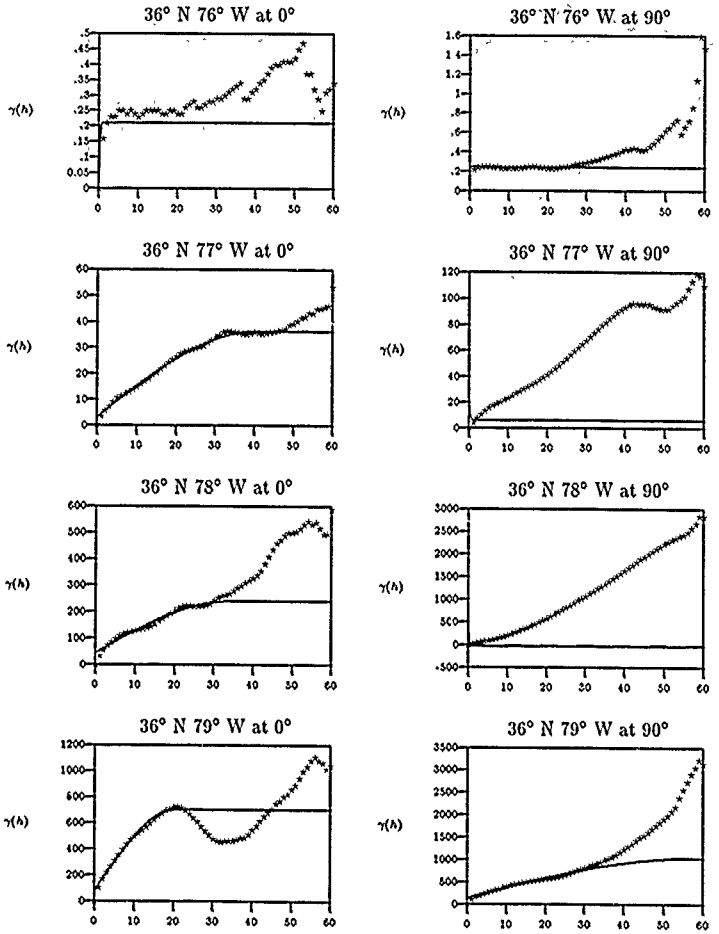

35° N 107° W at 90°
γ(h)
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
0 10 20 30 40 50 60

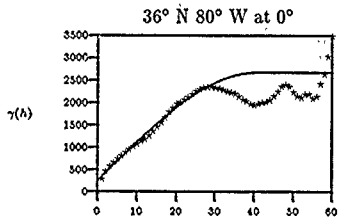

36° N 74° E at 0°
γ(h)
1.8e+06
1.6e+06
1.4e+06
1.2e+06
1e+06
800000
600000
400000
200000
0
0 10 20 30 40 50 60

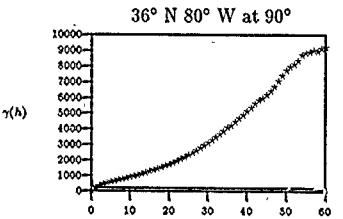

36° N 74° E at 90°
γ(h)
1.2e+06
1e+06
800000
600000
400000
200000
0
0 10 20 30 40 50 60

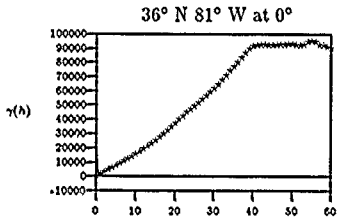

36° N 75° E at 0°
γ(h)
600000
500000
400000
300000
200000
100000
0
0 10 20 30 40 50 60

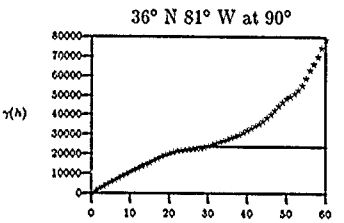

36° N 75° E at 90°
γ(h)
700000
600000
500000
400000
300000
200000
100000
0
0 10 20 30 40 50 60

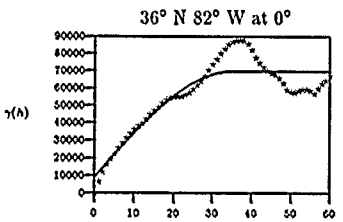

36° N 76° W at 0°
36° N 76° W at 90°
36° N 77° W at 0°
36° N 77° W at 90°
36° N 78° W at 0°
36° N 78° W at 90°
36° N 79° W at 0°
36° N 79° W at 90°
γ(h)

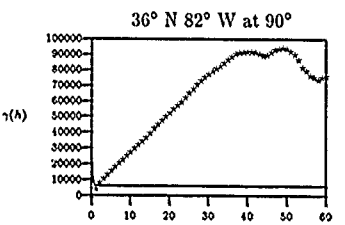

36° N 80° W at 0°
γ(h)
0
500
1000
1500
2000
2500
3000
3500
0 10 20 30 40 50 60

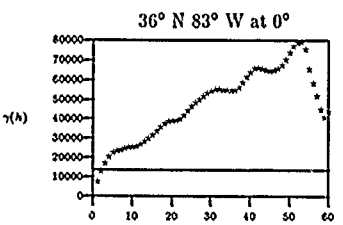

36° N 80° W at 90°
γ(h)
0
1000
2000
3000
4000
5000
6000
7000
8000
9000
10000
0 10 20 30 40 50 60

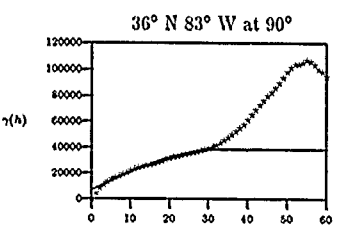

36° N 81° W at 0°
γ(h)
-10000
0
10000
20000
30000
40000
50000
60000
70000
80000
90000
100000
0 10 20 30 40 50 60

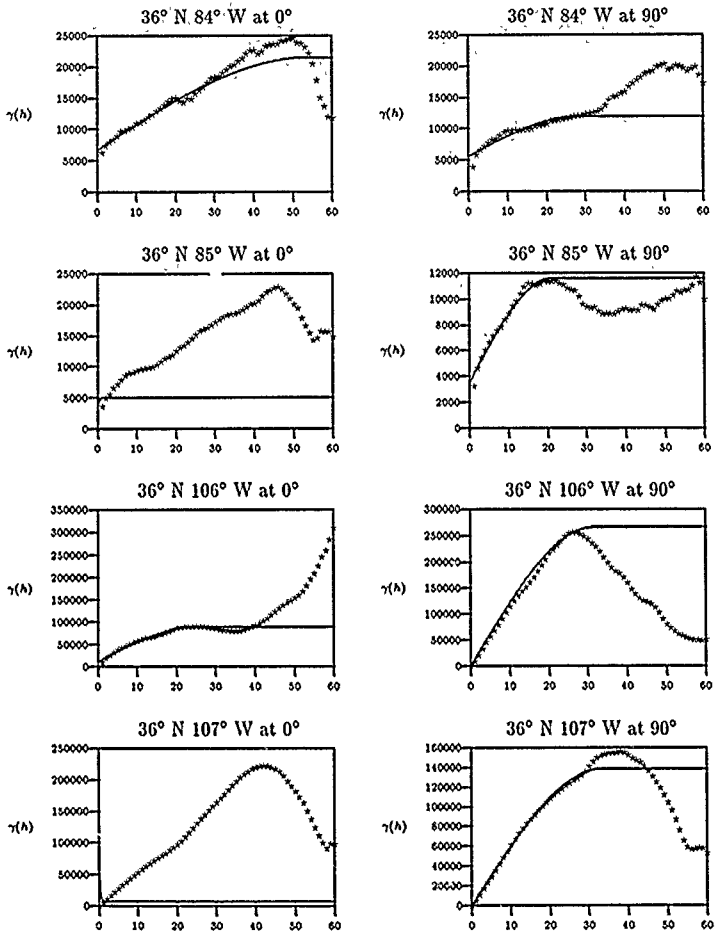

36° N 81° W at 90°
γ(h)
0
10000
20000
30000
40000
50000
60000
70000
80000
0 10 20 30 40 50 60

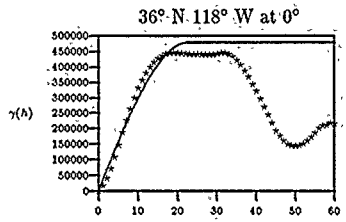

36° N 82° W at 0°
γ(h)
0
10000
20000
30000
40000
50000
60000
70000
80000
90000
0 10 20 30 40 50 60

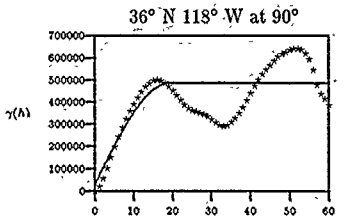

36° N 82° W at 90°
γ(h)
0
10000
20000
30000
40000
50000
60000
70000
80000
90000
100000
0 10 20 30 40 50 60

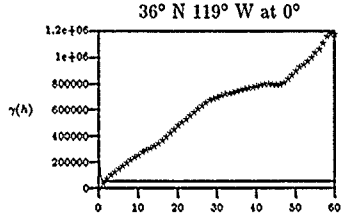

36° N 83° W at 0°
γ(h)
0
10000
20000
30000
40000
50000
60000
70000
80000
0 10 20 30 40 50 60

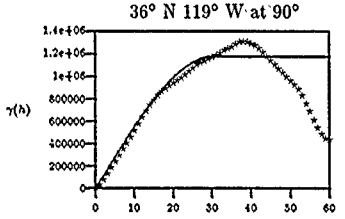

36° N 83° W at 90°
γ(h)
0
20000
40000
60000
80000
100000
120000
0 10 20 30 40 50 60

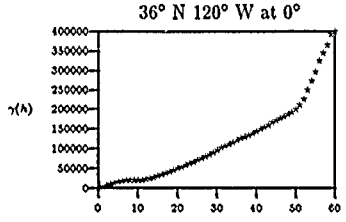

36° N 84° W at 0°
36° N 84° W at 90°
36° N 85° W at 0°
36° N 85° W at 90°
36° N 106° W at 0°
36° N 106° W at 90°
36° N 107° W at 0°
36° N 107° W at 90°
γ(h)

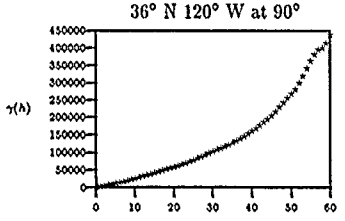

36° N 118° W at 0°
γ(h)
500000
450000
400000
350000
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

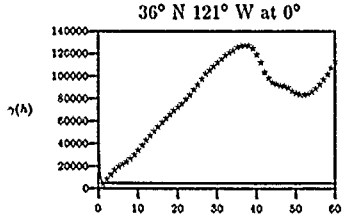

36° N 118° W at 90°
γ(h)
700000
600000
500000
400000
300000
200000
100000
0
0 10 20 30 40 50 60

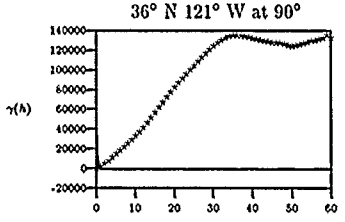

36° N 119° W at 0°
γ(h)
1.2e+06
1e+06
800000
600000
400000
200000
0
0 10 20 30 40 50 60

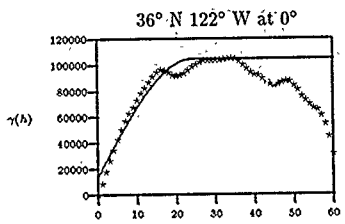

36° N 119° W at 90°
γ(h)
1.4e+06
1.2e+06
1e+06
800000
600000
400000
200000
0
0 10 20 30 40 50 60

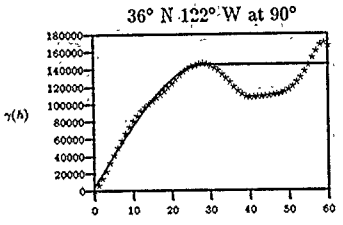

36° N 120° W at 0°
γ(h)
400000
350000
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

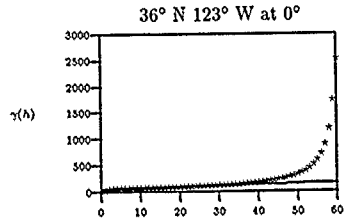

36° N 120° W at 90°
γ(h)
450000
400000
350000
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

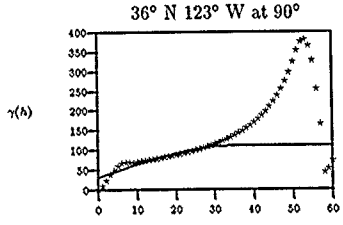

36° N 121° W at 0°
γ(h)
140000
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

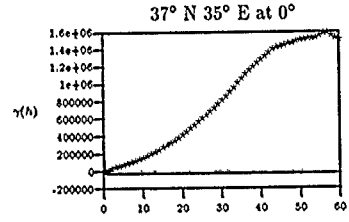

36° N 121° W at 90°
γ(h)
140000
120000
100000
60000
60000
40000
20000
0
-20000
0 10 20 30 40 50 60

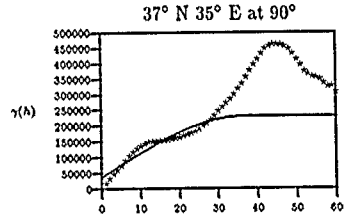

36° N 122° W at 0°
γ(h)
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

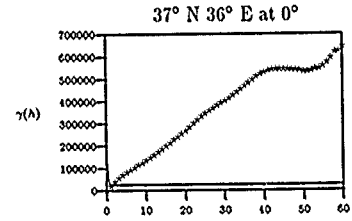

36° N 122° W at 90°
γ(h)
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

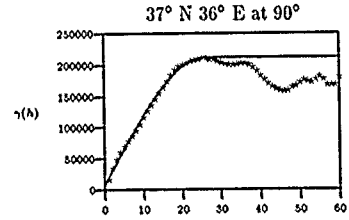

36° N 123° W at 0°
γ(h)
3000
2500
2000
1500
1000
500
0
0 10 20 30 40 50 60

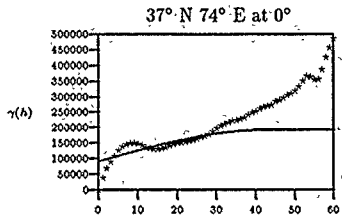

36° N 123° W at 90°
γ(h)
400
350
300
250
200
150
100
50
0
0 10 20 30 40 50 60

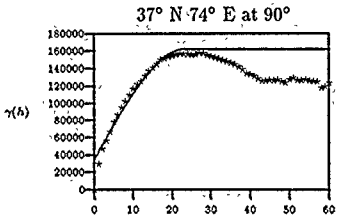

37° N 35° E at 0°
γ(h)
1.6e+06
1.4e+06
1.2e+06
1e+06
800000
600000
400000
200000
0
-200000
0 10 20 30 40 50 60

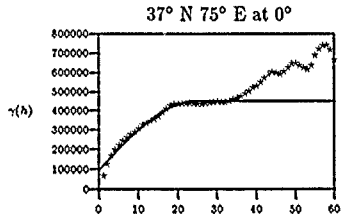

37° N 35° E at 90°
γ(h)
500000
450000
400000
350000
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

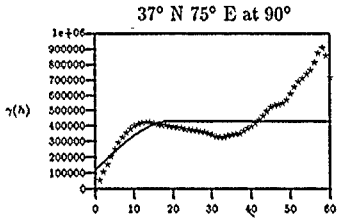

37° N 36° E at 0°
γ(h)
700000
600000
500000
400000
300000
200000
100000
0
0 10 20 30 40 50 60

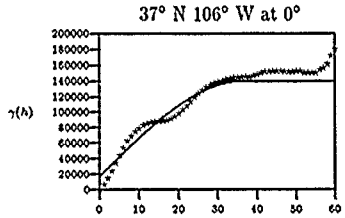

37° N 36° E at 90°
γ(h)
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

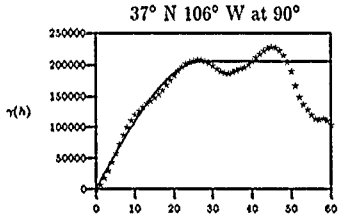

37° N 74° E at 0°
γ(h)
0
50000
100000
150000
200000
250000
300000
350000
400000
450000
0 10 20 30 40 50 60

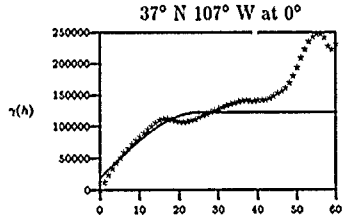

37° N 74° E at 90°
γ(h)
0
20000
40000
60000
80000
100000
120000
140000
160000
180000
0 10 20 30 40 50 60

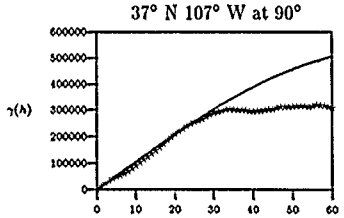

37° N 75° E at 0°
γ(h)
0
100000
200000
300000
400000
500000
600000
700000
800000
0 10 20 30 40 50 60

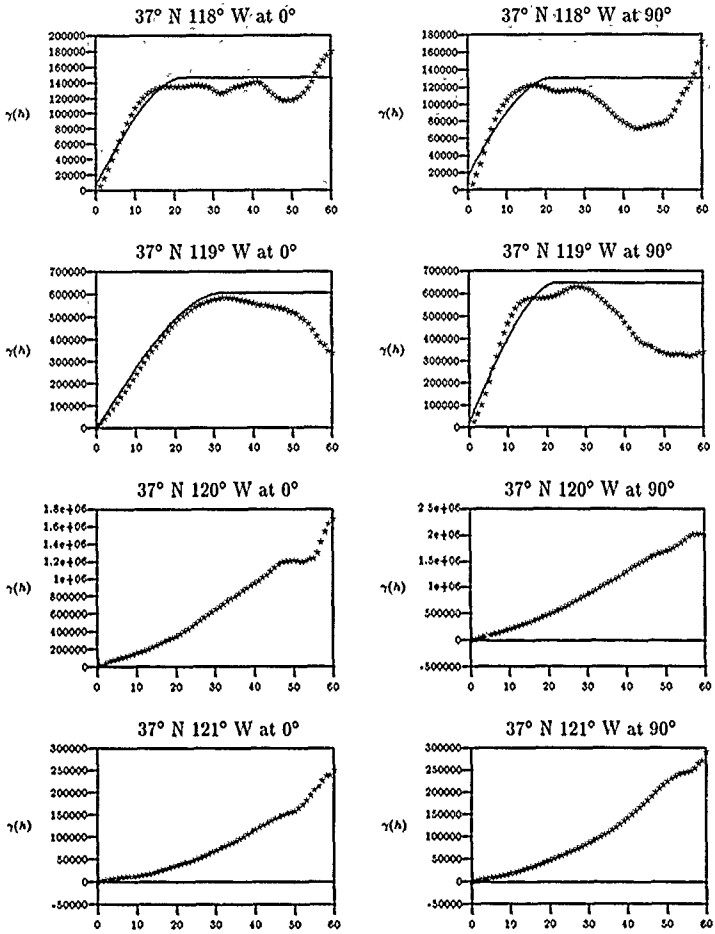

37° N 75° E at 90°
γ(h)
0
100000
200000
300000
400000
500000
600000
700000
800000
900000
1e+06
0 10 20 30 40 50 60

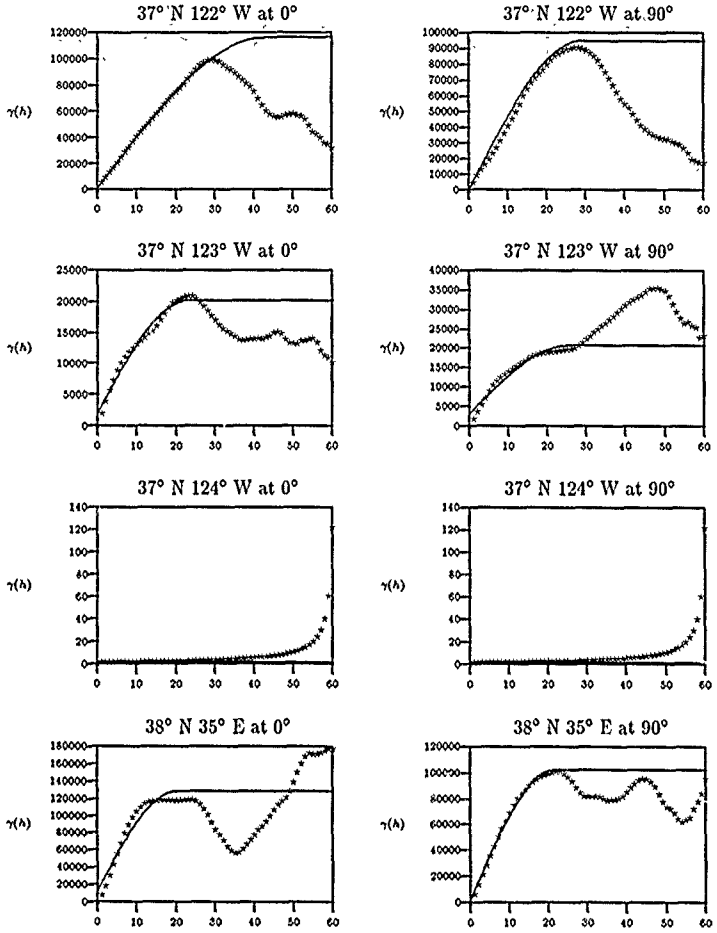

37° N 106° W at 0°
γ(h)
0
20000
40000
60000
80000
100000
120000
140000
160000
180000
200000
0 10 20 30 40 50 60


37° N 106° W at 90°
γ(h)
0
50000
100000
150000
200000
250000
0 10 20 30 40 50 60

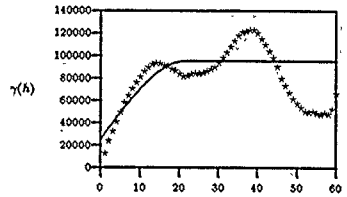

37° N 107° W at 0°
γ(h)
0
50000
100000
150000
200000
250000
0 10 20 30 40 50 60


37° N 107° W at 90°
γ(h)
0
100000
200000
300000
400000
500000
600000
0 10 20 30 40 50 60

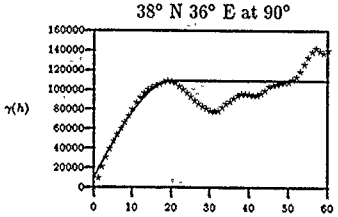

37° N 118° W at 0°
37° N 118° W at 90°
37° N 119° W at 0°
37° N 119° W at 90°
37° N 120° W at 0°
37° N 120° W at 90°
37° N 121° W at 0°
37° N 121° W at 90°
γ(h)


37° N 122° W at 0°
37° N 122° W at 90°
37° N 123° W at 0°
37° N 123° W at 90°
37° N 124° W at 0°
37° N 124° W at 90°
38° N 35° E at 0°
38° N 35° E at 90°
γ(h)

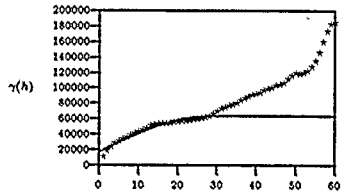
38° N 36° E at 0°



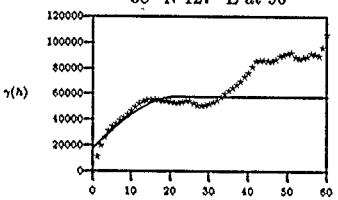
38° N 36° E at 90°



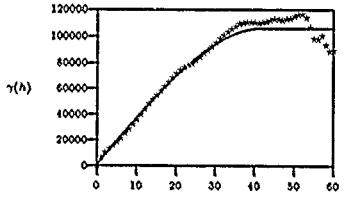
38° N 127° E at 0°



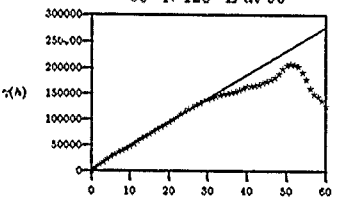
38° N 127° E at 90°



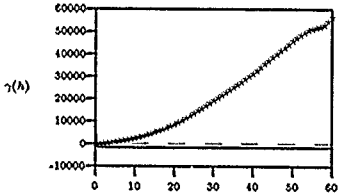
38° N 128° E at 0°



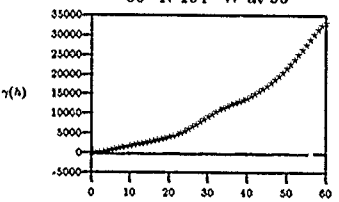
38° N 128° E at 90°

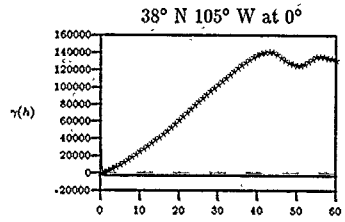


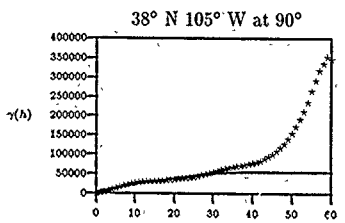
38° N 104° W at 0°

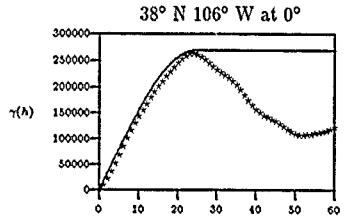


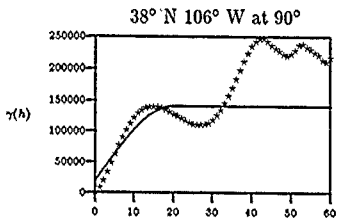
38° N 104° W at 90°

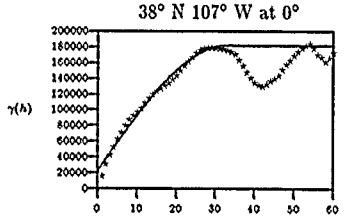


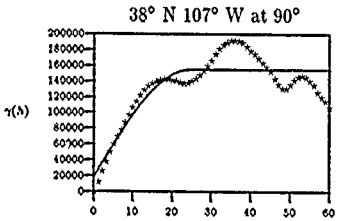

38° N 105° W at 0°
γ(h)
160000
140000
120000
100000
80000
60000
40000
20000
0
-20000
0 10 20 30 40 50 60

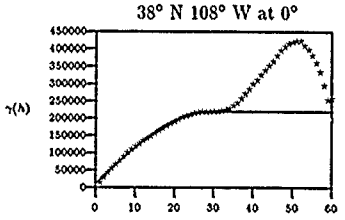

38° N 105° W at 90°
γ(h)
400000
350000
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

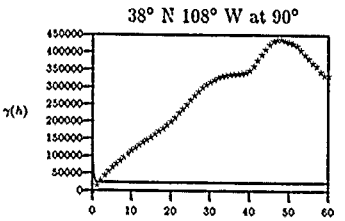

38° N 106° W at 0°
γ(h)
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

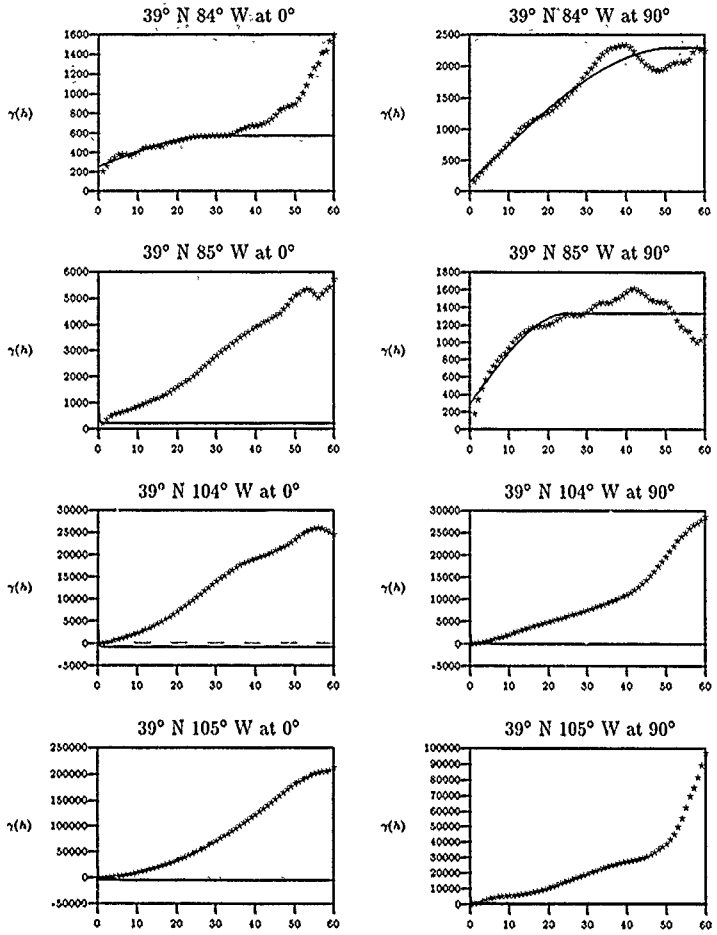

38° N 106° W at 90°
γ(h)
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

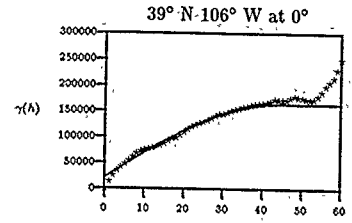

38° N 107° W at 0°
γ(h)
200000
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

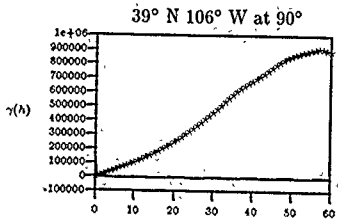

38° N 107° W at 90°
γ(h)
200000
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

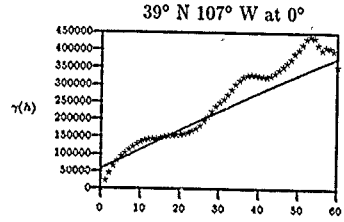

38° N 108° W at 0°
γ(h)
450000
400000
350000
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

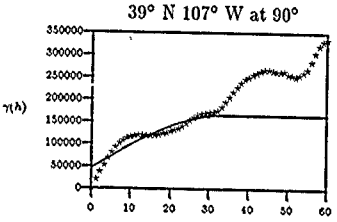

38° N 108° W at 90°
γ(h)
450000
400000
350000
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

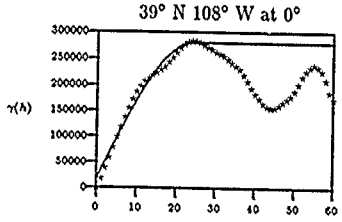

39° N 84° W at 0°
39° N 84° W at 90°
39° N 85° W at 0°
39° N 85° W at 90°
39° N 104° W at 0°
39° N 104° W at 90°
39° N 105° W at 0°
39° N 105° W at 90°
γ(h)

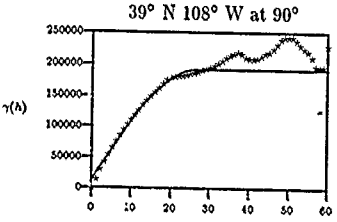

39° N 106° W at 0°
γ(h)
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

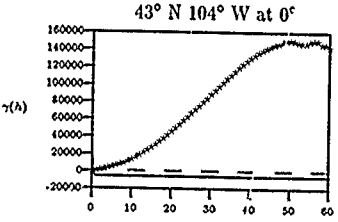

39° N 106° W at 90°
γ(h)
1e+06
900000
800000
700000
600000
500000
400000
300000
200000
100000
0
-100000
0 10 20 30 40 50 60

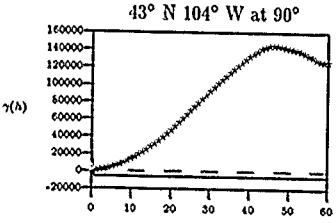

39° N 107° W at 0°
γ(h)
450000
400000
350000
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

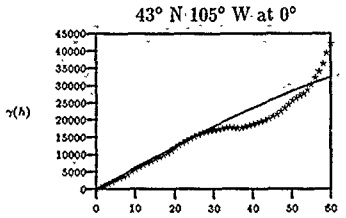

39° N 107° W at 90°
γ(h)
350000
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

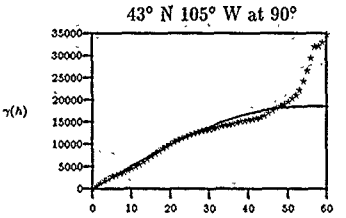

39° N 108° W at 0°
γ(h)
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

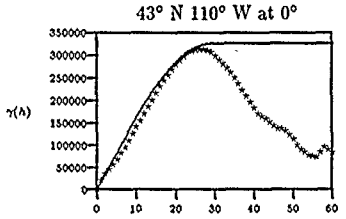

39° N 108° W at 90°
γ(h)
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

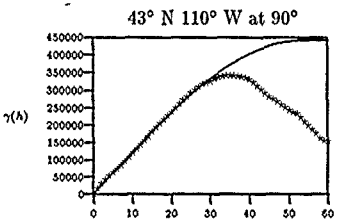

43° N 104° W at 0°
γ(h)
160000
140000
120000
100000
80000
60000
40000
20000
0
-20000
0 10 20 30 40 50 60

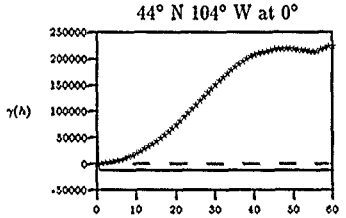

43° N 104° W at 90°
γ(h)
160000
140000
120000
100000
80000
60000
40000
20000
0
-20000
0 10 20 30 40 50 60

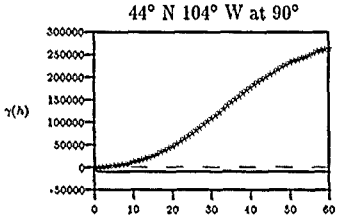

43° N 105° W at 0°
γ(h)
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
0 10 20 30 40 50 60

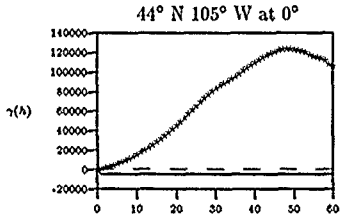

43° N 105° W at 90°
γ(h)
35000
30000
25000
20000
15000
10000
5000
0
0 10 20 30 40 50 60

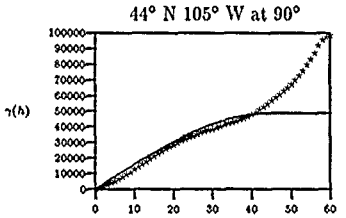

43° N 110° W at 0°
γ(h)
350000
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

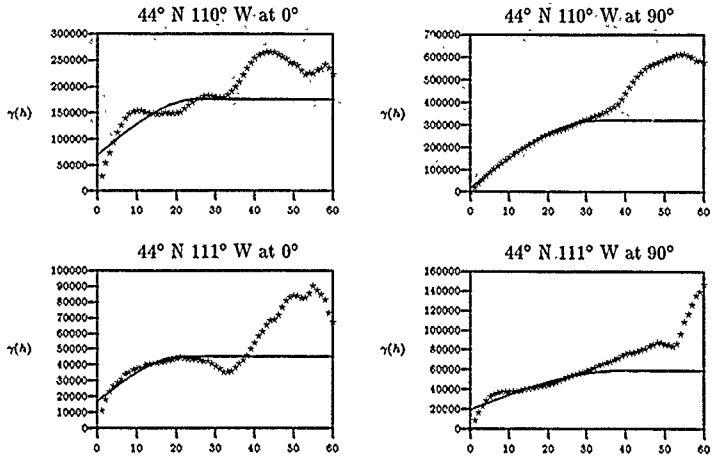

43° N 110° W at 90°
γ(h)
450000
400000
350000
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60


44° N 104° W at 0°
γ(h)
250000
200000
150000
100000
50000
0
-50000
0 10 20 30 40 50 60


44° N 104° W at 90°
γ(h)
300000
250000
200000
150000
100000
50000
0
-50000
0 10 20 30 40 50 60


44° N 105° W at 0°
γ(h)
140000
120000
100000
80000
60000
40000
20000
0
-20000
0 10 20 30 40 50 60

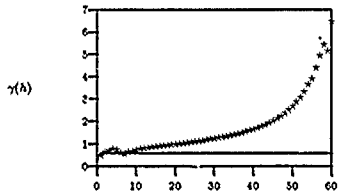

44° N 105° W at 90°
γ(h)
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
0 10 20 30 40 50 60


44° N 110° W at 0°
γ(h)
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60
44° N 110° W at 90°
γ(h)
700000
600000
500000
400000
300000
200000
100000
0
0 10 20 30 40 50 60
44° N 111° W at 0°
γ(h)
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
0 10 20 30 40 50 60
44° N 111° W at 90°
γ(h)
160000
140000
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

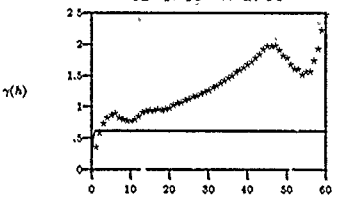
C.2 Variograms of the Residuals from the DTED Evaluation Cells

The following pages show directional experimental variograms and spherical variogram models for the DTED Evaluation Cells listed in Chapter III. These variograms were generated by `varfit` on *residual* terrain elevation data from these cells; the global trend was removed using `resid`. The latitude and longitude of the southwest corner of the DTED cell used to generate each variogram is provided above the variogram. Also indicated is the angle at which the experimental variogram was generated (in degrees clockwise from north). The experimental variograms are denoted by the stars ($\star$), while the spherical variogram models are denoted by solid lines.

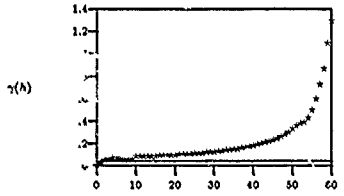
32° N 80° W at 0°



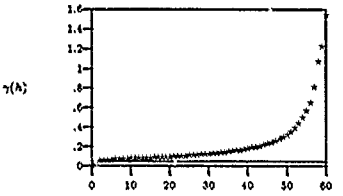
32° N 80° W at 90°



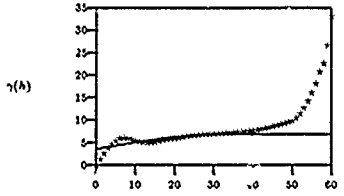
33° N 78° W at 0°



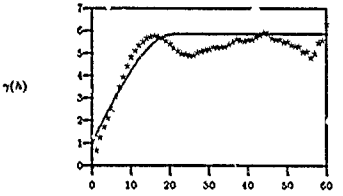
33° N 78° W at 90°



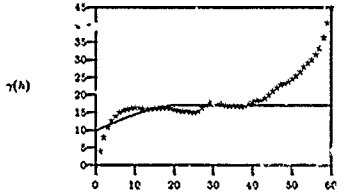
33° N 79° W at 0°



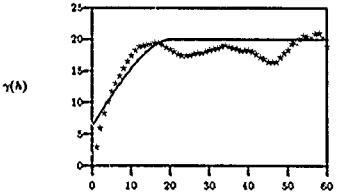
33° N 79° W at 90°

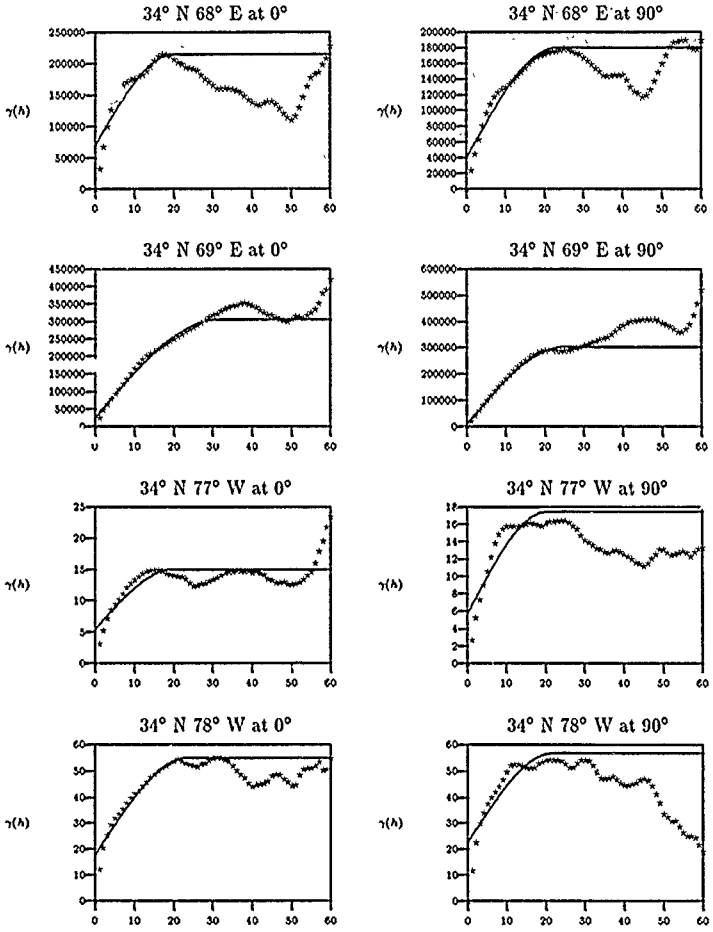


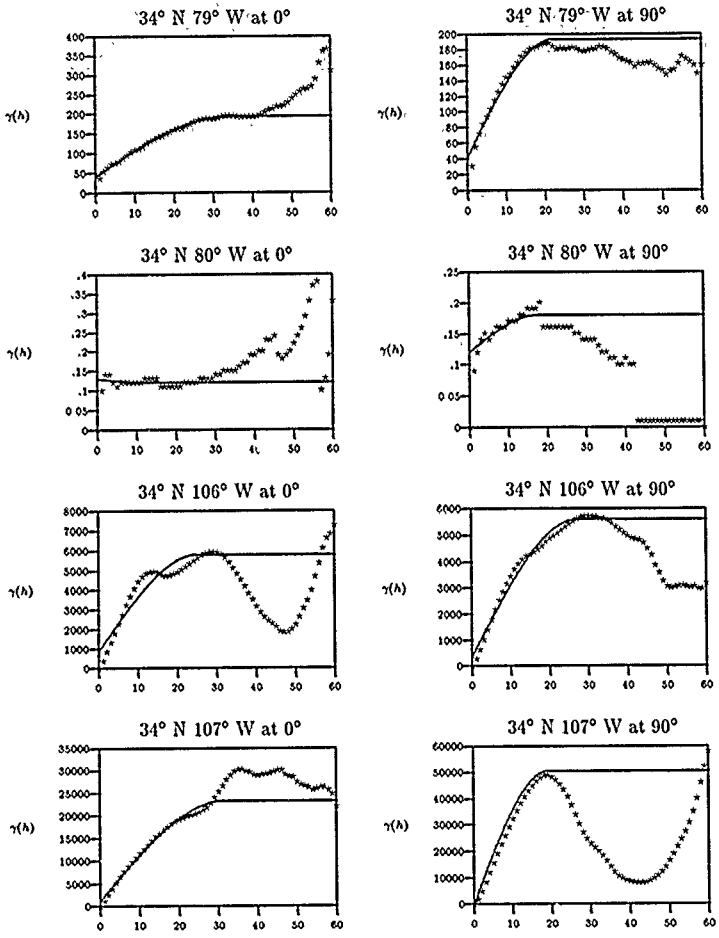
33° N 80° W at 0°

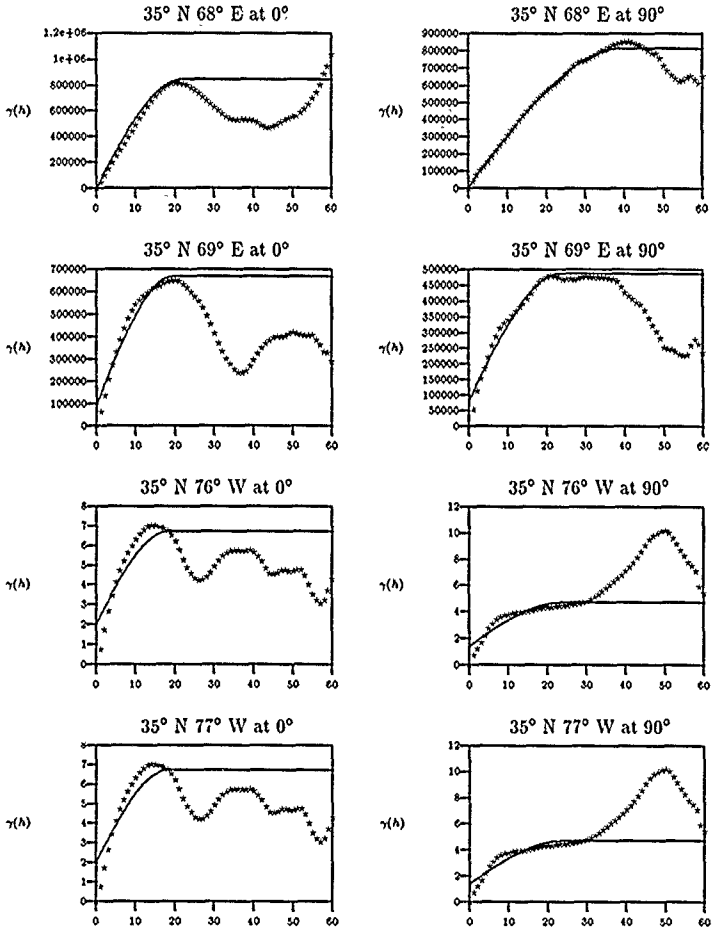


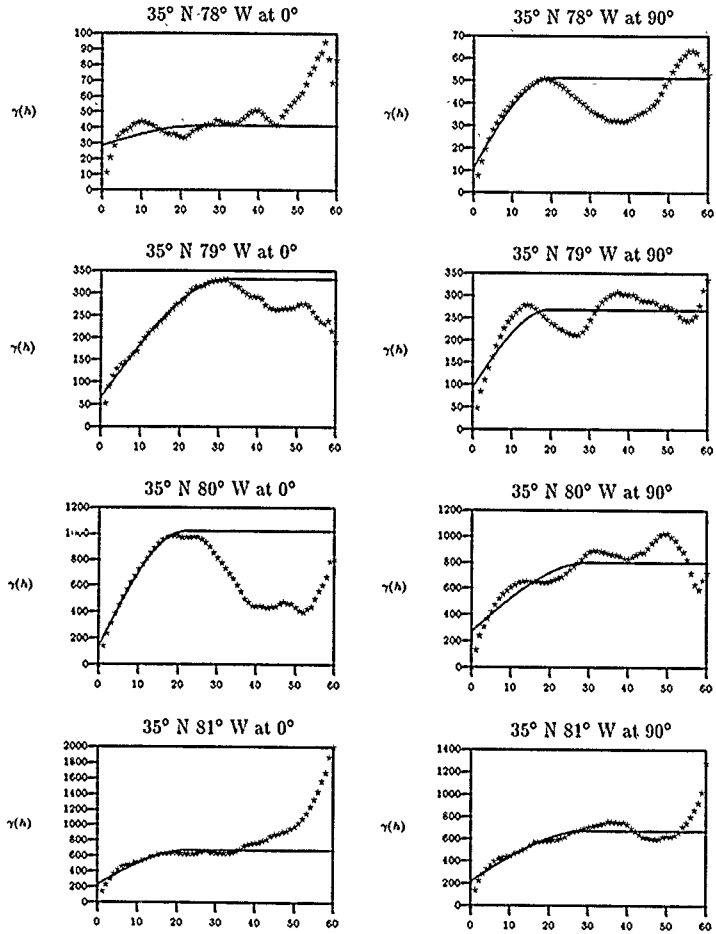
33° N 80° W at 90°

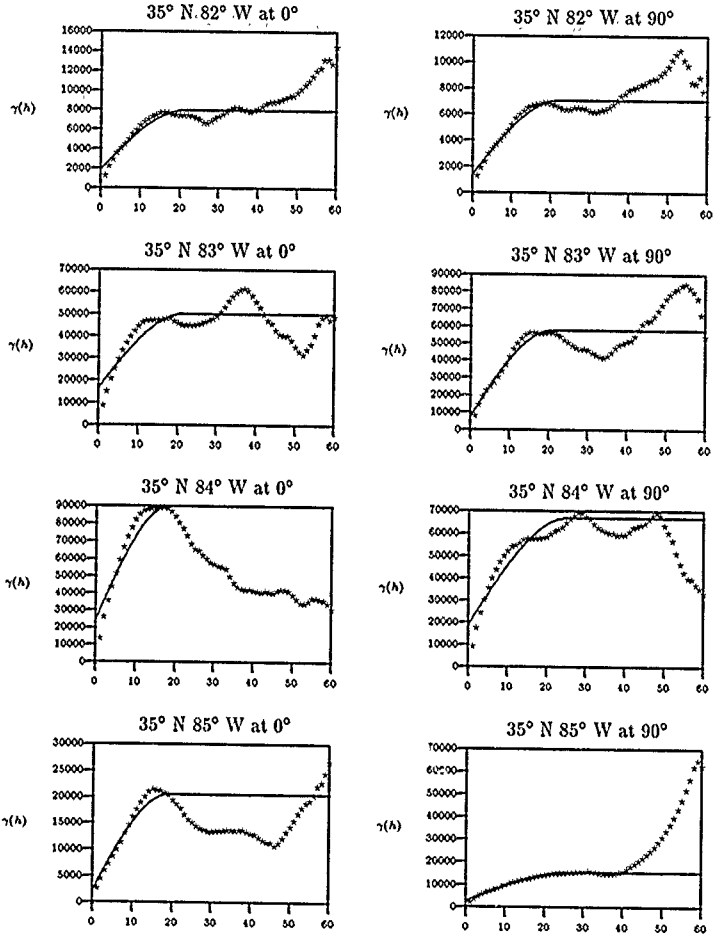


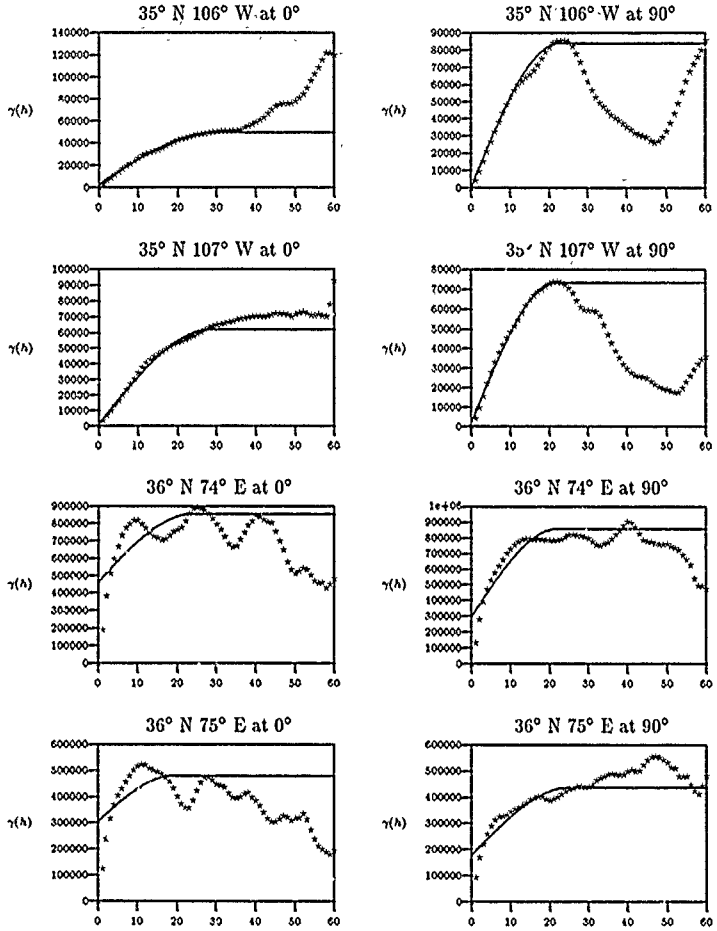

34° N 68° E at 0°
34° N 68° E at 90°
34° N 69° E at 0°
34° N 69° E at 90°
34° N 77° W at 0°
34° N 77° W at 90°
34° N 78° W at 0°
34° N 78° W at 90°
γ(h)

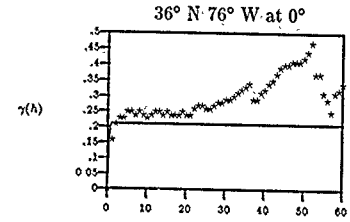

34° N 79° W at 0°
34° N 79° W at 90°
34° N 80° W at 0°
34° N 80° W at 90°
34° N 106° W at 0°
34° N 106° W at 90°
34° N 107° W at 0°
34° N 107° W at 90°
γ(h)

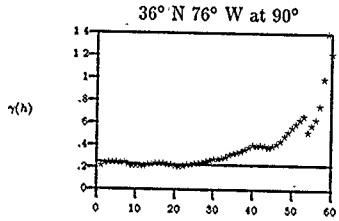

35° N 68° E at 0°
35° N 68° E at 90°
35° N 69° E at 0°
35° N 69° E at 90°
35° N 76° W at 0°
35° N 76° W at 90°
35° N 77° W at 0°
35° N 77° W at 90°
γ(h)

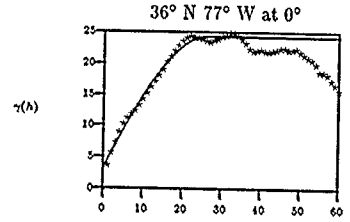

35° N 78° W at 0°
35° N 78° W at 90°
35° N 79° W at 0°
35° N 79° W at 90°
35° N 80° W at 0°
35° N 80° W at 90°
35° N 81° W at 0°
35° N 81° W at 90°
γ(h)

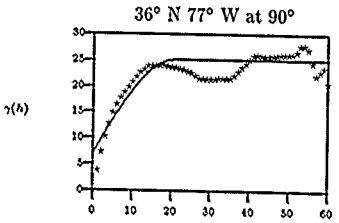

35° N 82° W at 0°
35° N 82° W at 90°
35° N 83° W at 0°
35° N 83° W at 90°
35° N 84° W at 0°
35° N 84° W at 90°
35° N 85° W at 0°
35° N 85° W at 90°
γ(h)

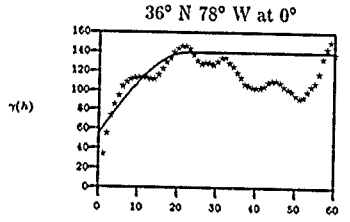

35° N 106° W at 0°
35° N 106° W at 90°
35° N 107° W at 0°
35° N 107° W at 90°
36° N 74° E at 0°
36° N 74° E at 90°
36° N 75° E at 0°
36° N 75° E at 90°
γ(h)

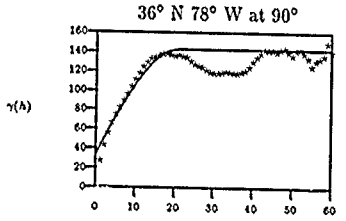

36° N 76° W at 0°
γ(h)

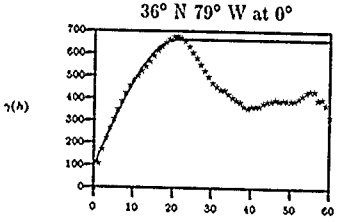

36° N 76° W at 90°
γ(h)

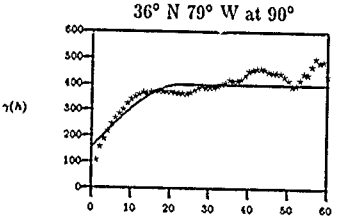

36° N 77° W at 0°
γ(h)

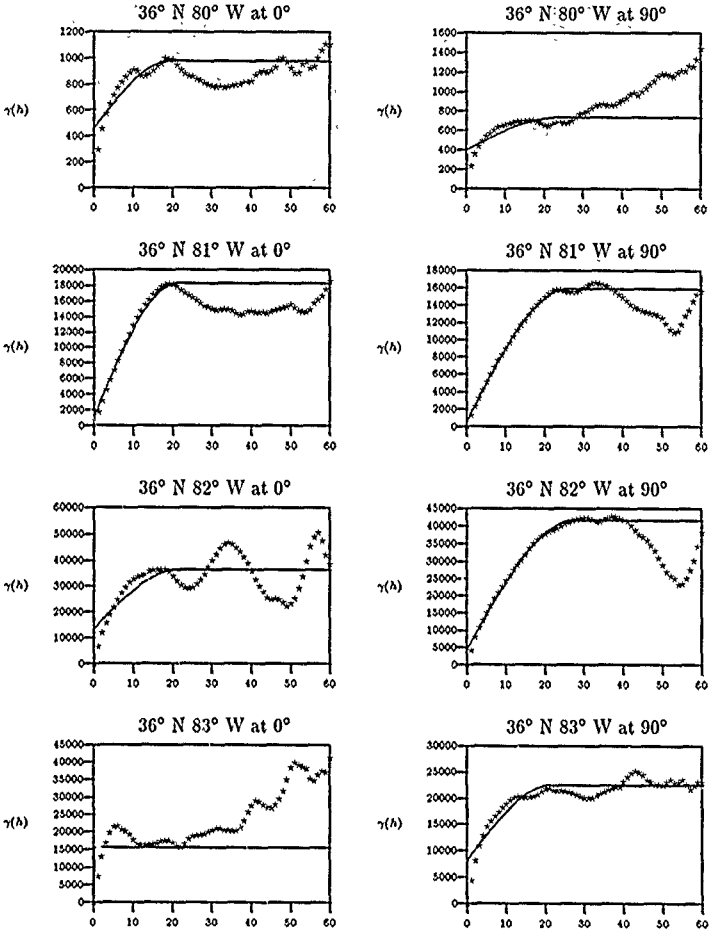

36° N 77° W at 90°
γ(h)

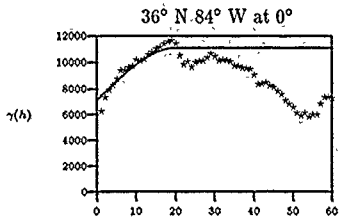

36° N 78° W at 0°
γ(h)

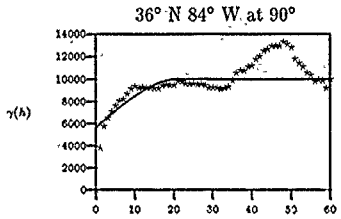

36° N 78° W at 90°
γ(h)

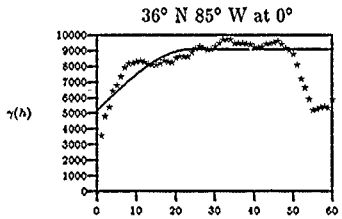

36° N 79° W at 0°
γ(h)

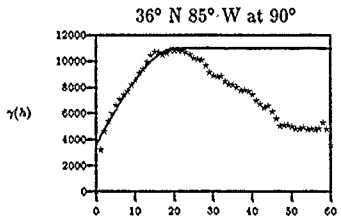

36° N 79° W at 90°
γ(h)

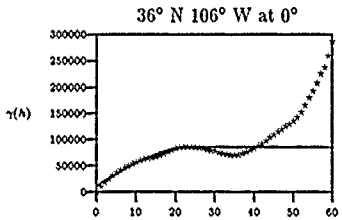

36° N 80° W at 0°
36° N 80° W at 90°
36° N 81° W at 0°
36° N 81° W at 90°
36° N 82° W at 0°
36° N 82° W at 90°
36° N 83° W at 0°
36° N 83° W at 90°
γ(h)

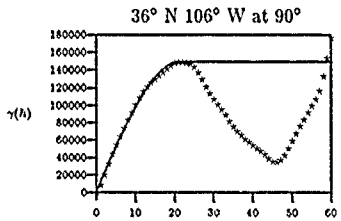

36° N 84° W at 0°
γ(h)
12000
10000
8000
6000
4000
2000
0
0 10 20 30 40 50 60

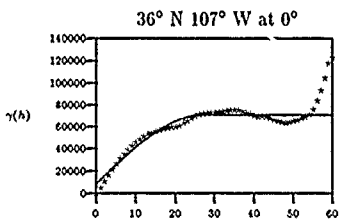

36° N 84° W at 90°
γ(h)
14000
12000
10000
8000
6000
4000
2000
0
0 10 20 30 40 50 60

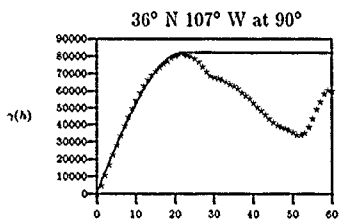

36° N 85° W at 0°
γ(h)
10000
9000
8000
7000
6000
5000
4000
3000
2000
1000
0
0 10 20 30 40 50 60

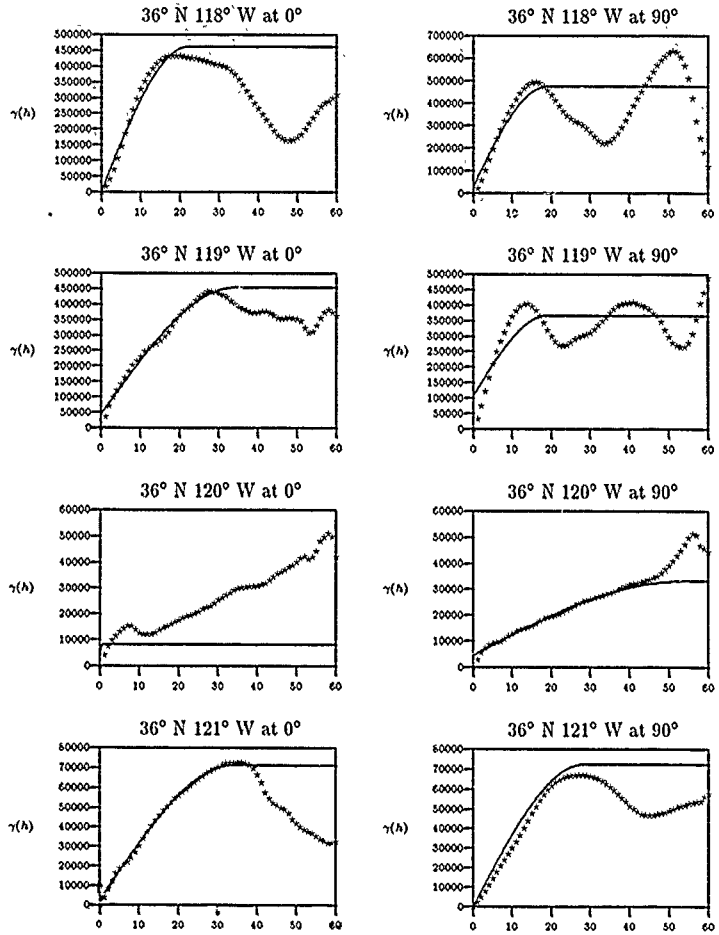

36° N 85° W at 90°
γ(h)
12000
10000
8000
6000
4000
2000
0
0 10 20 30 40 50 60

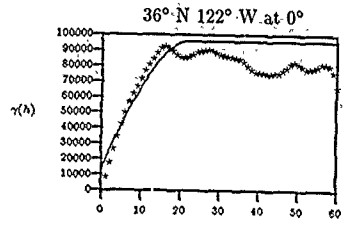

36° N 106° W at 0°
γ(h)
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

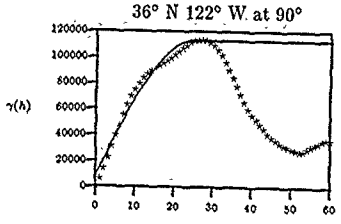

36° N 106° W at 90°
γ(h)
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

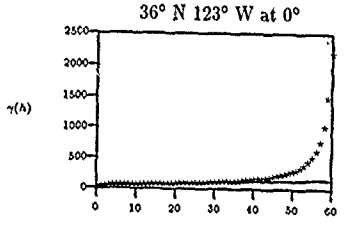

36° N 107° W at 0°
γ(h)
140000
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

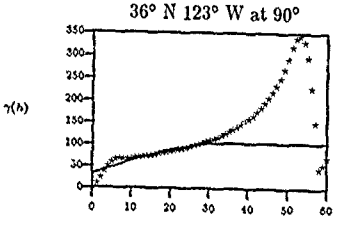

36° N 107° W at 90°
γ(h)
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
0 10 20 30 40 50 60

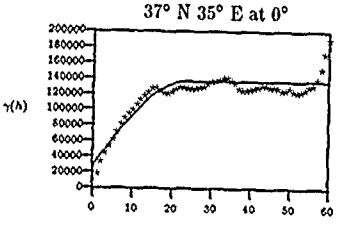

36° N 118° W at 0°
36° N 118° W at 90°
36° N 119° W at 0°
36° N 119° W at 90°
36° N 120° W at 0°
36° N 120° W at 90°
36° N 121° W at 0°
36° N 121° W at 90°
γ(h)

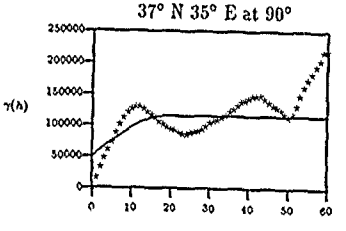

36° N 122° W at 0°
γ(h)

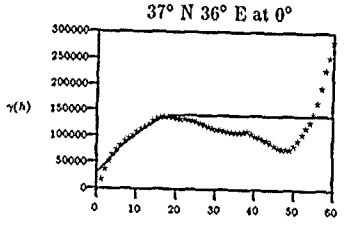

36° N 122° W at 90°
γ(h)

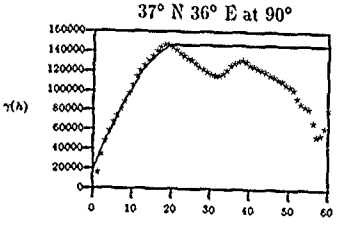

36° N 123° W at 0°
γ(h)

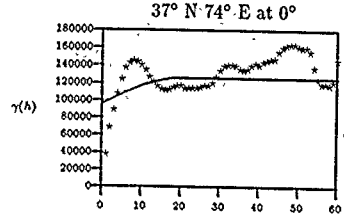

36° N 123° W at 90°
γ(h)

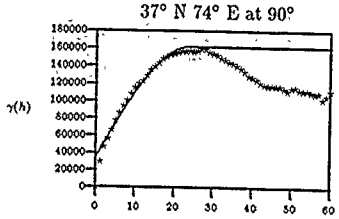

37° N 35° E at 0°
γ(h)

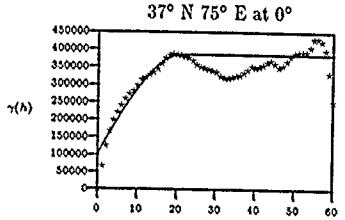

37° N 35° E at 90°
γ(h)

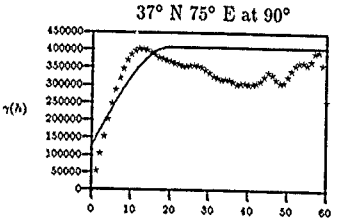

37° N 36° E at 0°
γ(h)

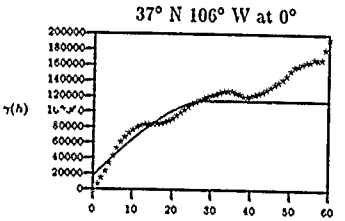

37° N 36° E at 90°
γ(h)

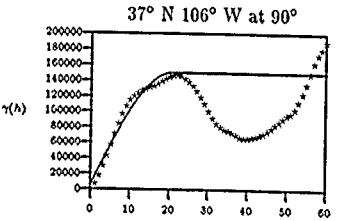

37° N 74° E at 0°
γ(h)
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

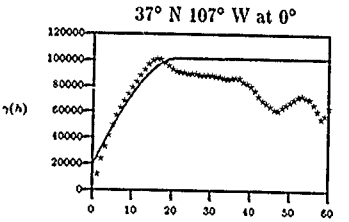

37° N 74° E at 90°
γ(h)
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

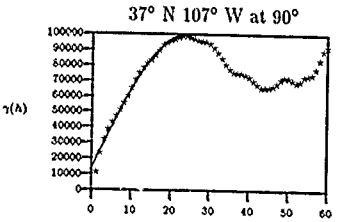

37° N 75° E at 0°
γ(h)
450000
400000
350000
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

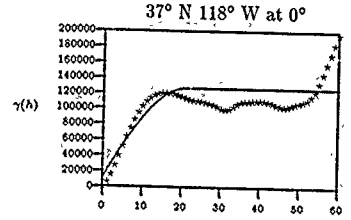

37° N 75° E at 90°
γ(h)
450000
400000
350000
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

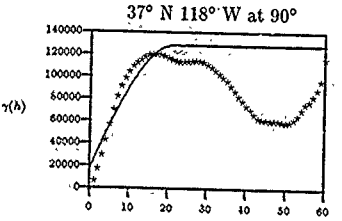

37° N 106° W at 0°
γ(h)
200000
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

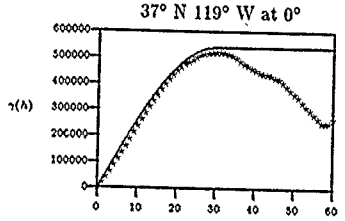

37° N 106° W at 90°
γ(h)
200000
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

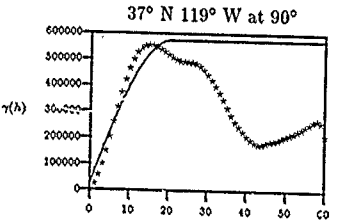

37° N 107° W at 0°
γ(h)
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

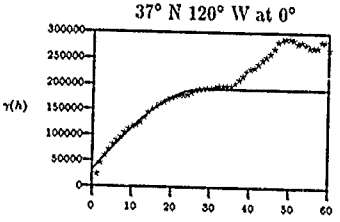

37° N 107° W at 90°
γ(h)
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
0 10 20 30 40 50 60

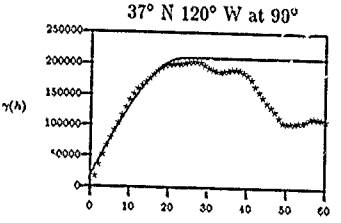

37° N 118° W at 0°
γ(h)
200000
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

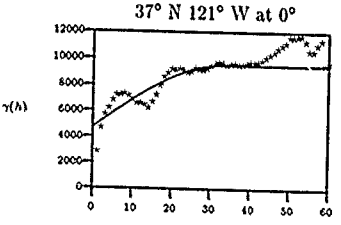

37° N 118° W at 90°
γ(h)
140000
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

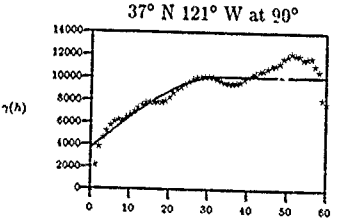

37° N 119° W at 0°
γ(h)
600000
500000
400000
300000
200000
100000
0
0 10 20 30 40 50 60

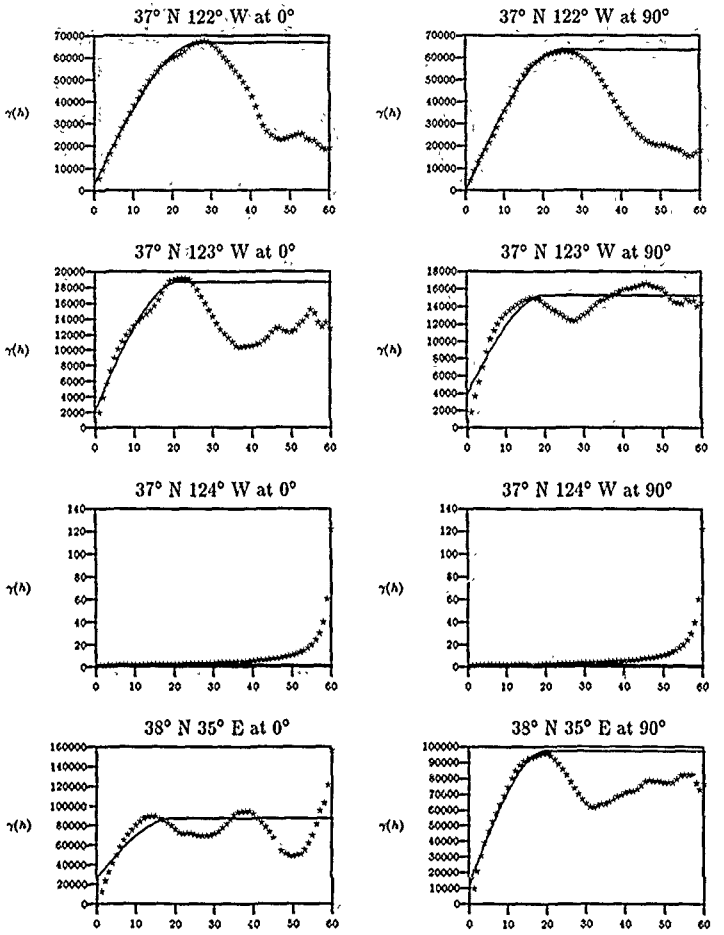

37° N 119° W at 90°
γ(h)
600000
500000
400000
300000
200000
100000
0
0 10 20 30 40 50 60

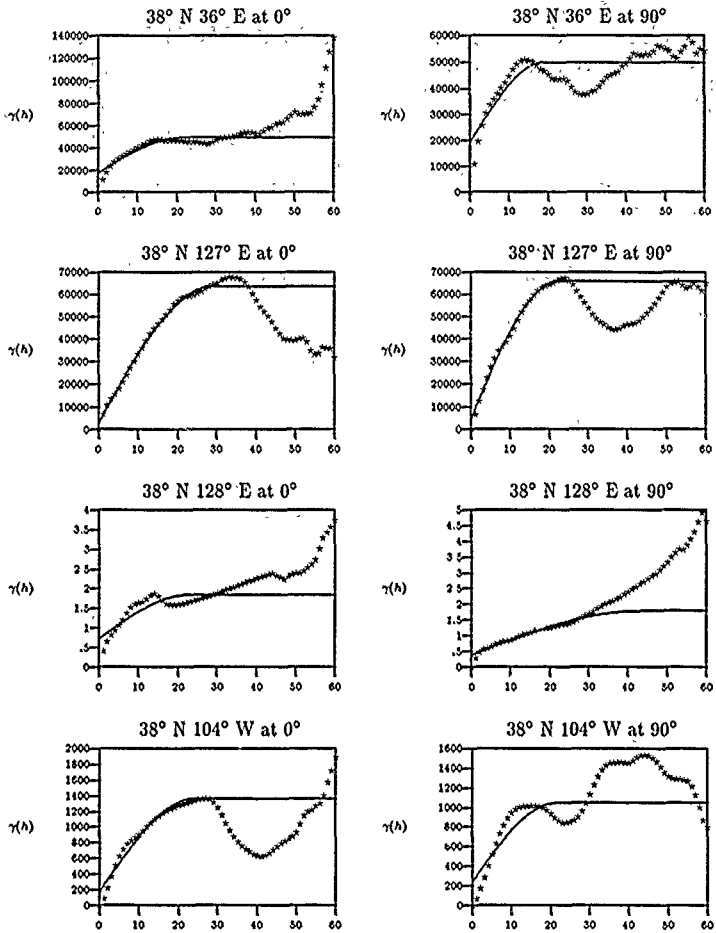

37° N 120° W at 0°
γ(h)
300000
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

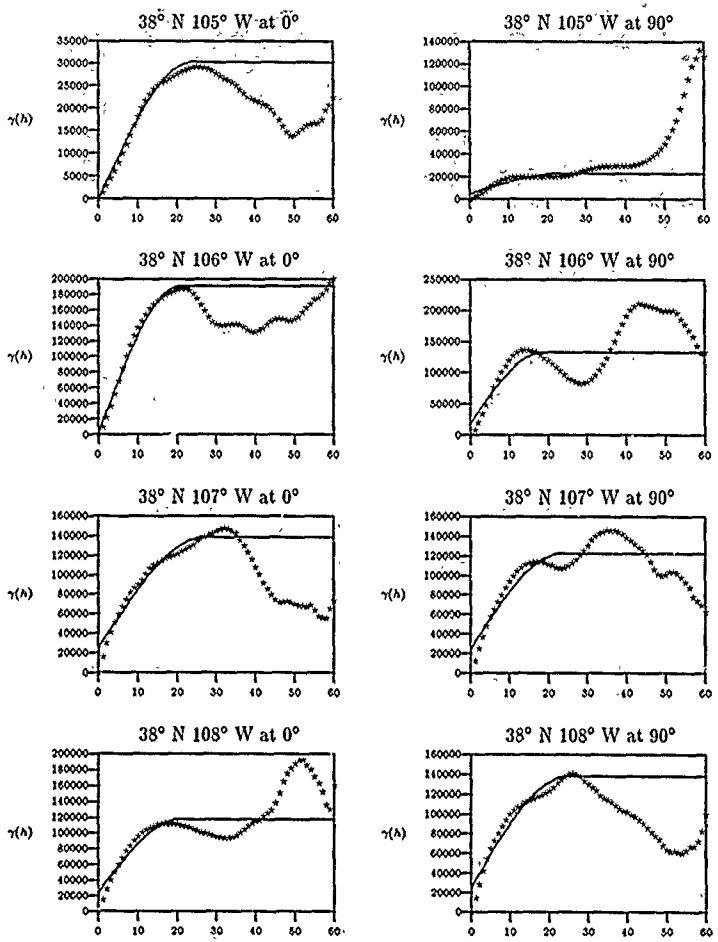

37° N 120° W at 90°
γ(h)
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60

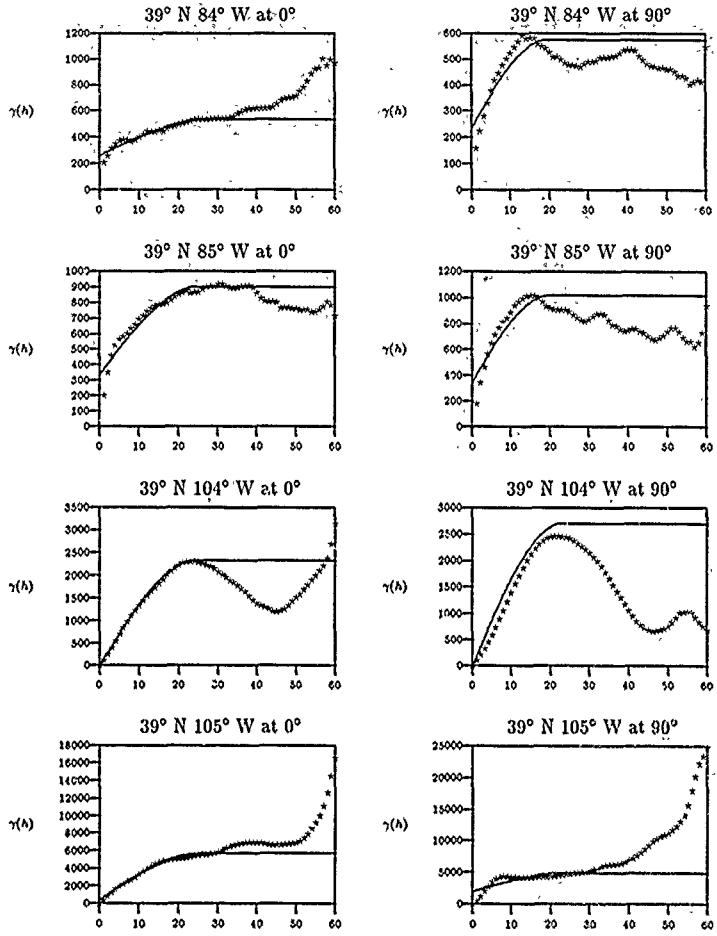

37° N 121° W at 0°
γ(h)
12000
10000
8000
6000
4000
2000
0
0 10 20 30 40 50 60

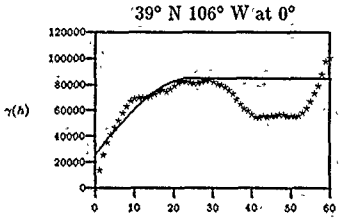

37° N 121° W at 90°
γ(h)
14000
12000
10000
8000
6000
4000
2000
0
0 10 20 30 40 50 60

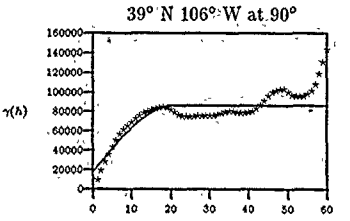

37° N 122° W at 0°
37° N 122° W at 90°
37° N 123° W at 0°
37° N 123° W at 90°
37° N 124° W at 0°
37° N 124° W at 90°
38° N 35° E at 0°
38° N 35° E at 90°
γ(h)

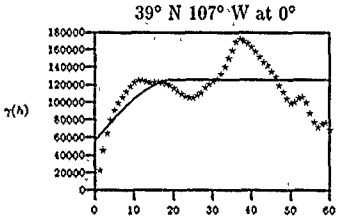

38° N 36° E at 0°
38° N 36° E at 90°
38° N 127° E at 0°
38° N 127° E at 90°
38° N 128° E at 0°
38° N 128° E at 90°
38° N 104° W at 0°
38° N 104° W at 90°
γ(h)

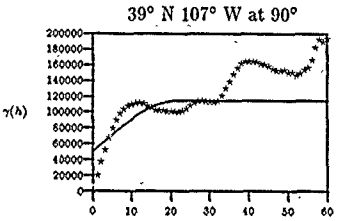

38° N 105° W at 0°
38° N 105° W at 90°
38° N 106° W at 0°
38° N 106° W at 90°
38° N 107° W at 0°
38° N 107° W at 90°
38° N 108° W at 0°
38° N 108° W at 90°
γ(h)

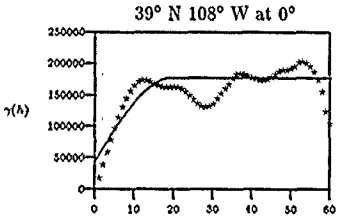

39° N 84° W at 0°
39° N 84° W at 90°
39° N 85° W at 0°
39° N 85° W at 90°
39° N 104° W at 0°
39° N 104° W at 90°
39° N 105° W at 0°
39° N 105° W at 90°
γ(h)

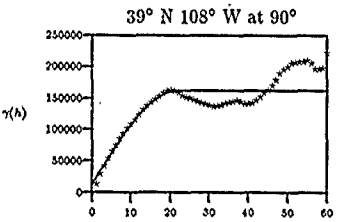

39° N 106° W at 0°
γ(h)
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

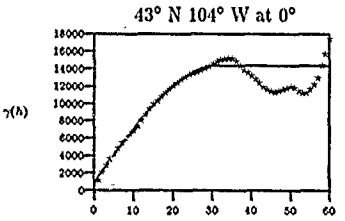

39° N 106° W at 90°
γ(h)
160000
140000
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

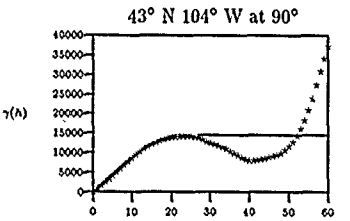

39° N 107° W at 0°
γ(h)
160000
140000
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

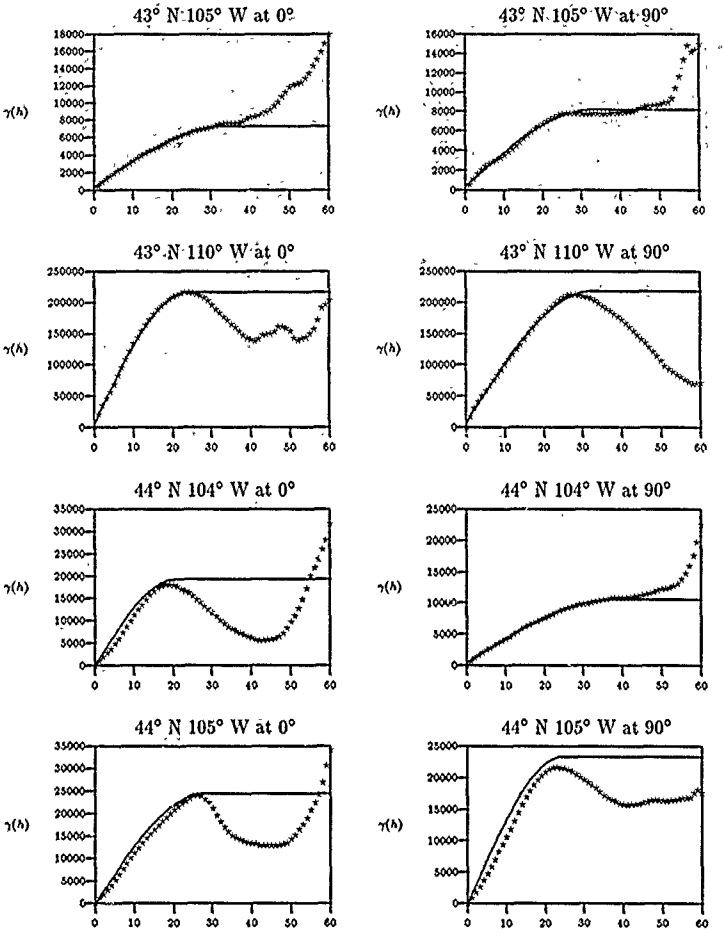

39° N 107° W at 90°
γ(h)
200000
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
0 10 20 30 40 50 60

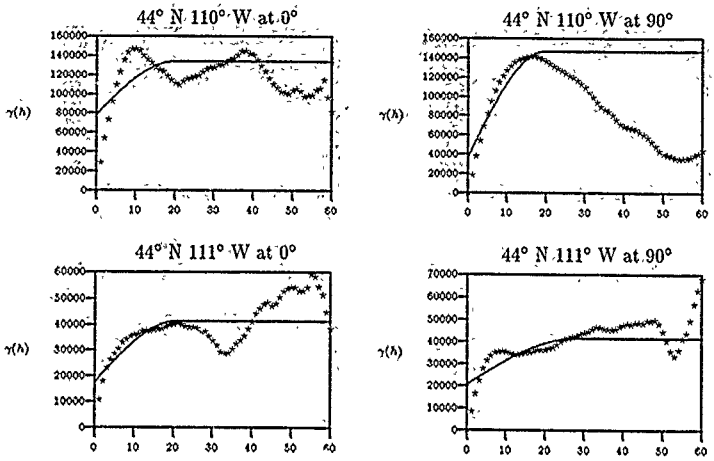

39° N 108° W at 0°
γ(h)
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60


39° N 108° W at 90°
γ(h)
250000
200000
150000
100000
50000
0
0 10 20 30 40 50 60


43° N 104° W at 0°
γ(h)
18000
16000
14000
12000
10000
8000
6000
4000
2000
0
0 10 20 30 40 50 60

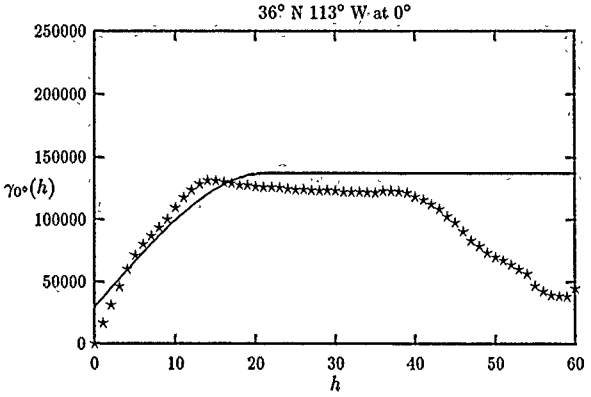

43° N 104° W at 90°
γ(h)
40000
35000
30000
25000
20000
15000
10000
5000
0
0 10 20 30 40 50 60


43° N 105° W at 0°
43° N 105° W at 90°
43° N 110° W at 0°
43° N 110° W at 90°
44° N 104° W at 0°
44° N 104° W at 90°
44° N 105° W at 0°
44° N 105° W at 90°
γ(h)

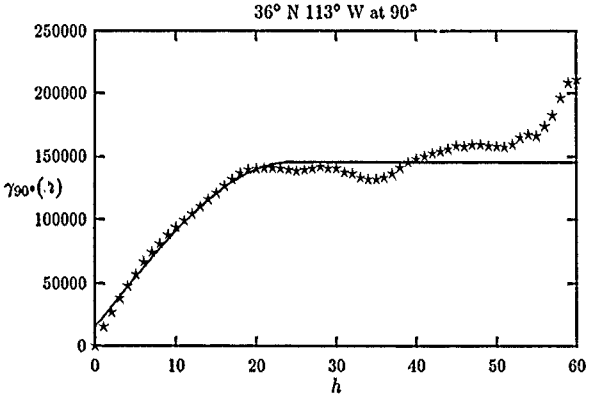

44° N 110° W at 0°
44° N 110° W at 90°
44° N 111° W at 0°
44° N 111° W at 90°
γ(λ)

C.3 Variograms for DTED Test Cells

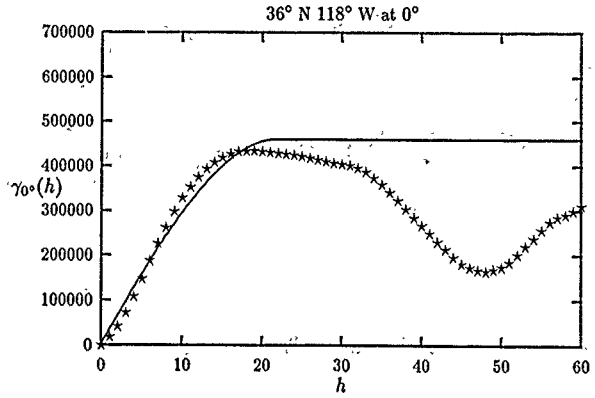
The following pages show directional experimental variograms and spherical variogram models for the DTED Test Cells listed in Chapter V. These experimental variograms and variogram models were generated by `varfit` on *residual* terrain elevation data from these cells; the global trend was removed using `resid`. The latitude and longitude of the southwest corner of the DTED cell used to generate each variogram is provided above the variogram. Also indicated is the angle at which the experimental variogram was generated (in degrees clockwise from north) and the equation for the spherical model fitted to the experimental variogram by `varfit`. The experimental variograms are denoted by the stars ($\star$), while the spherical variogram models are denoted by solid lines.



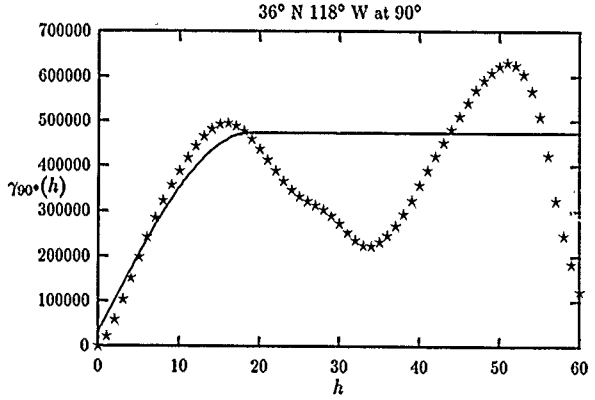
$$\gamma_{0°}(h) = 107291.45\left[\frac{3}{2}\left(\frac{h}{21.29}\right) - \frac{1}{2}\left(\frac{h}{21.29}\right)^3\right] + 29852.85$$



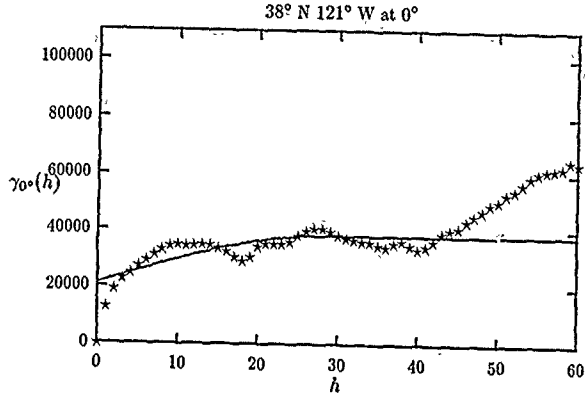
$$\gamma_{90°}(h) = 129255.93\left[\frac{3}{2}\left(\frac{h}{24.10}\right) - \frac{1}{2}\left(\frac{h}{24.10}\right)^3\right] + 15659.18$$



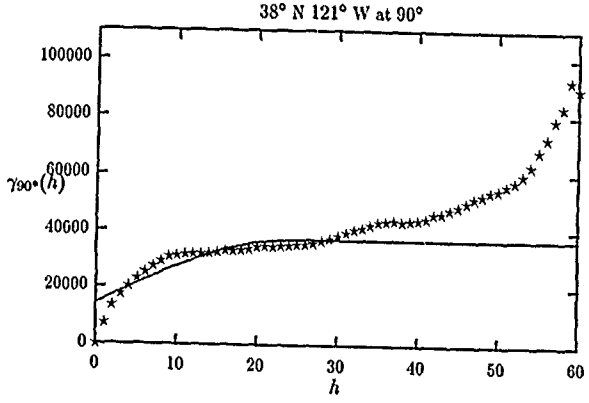
$$\gamma_{0°}(h) = 456047.69\left[\frac{3}{2}\left(\frac{h}{21.86}\right) - \frac{1}{2}\left(\frac{h}{21.86}\right)^3\right] + 4740.00$$



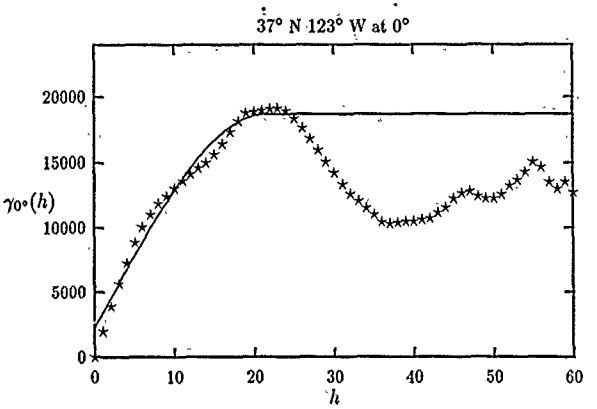
$$\gamma_{90°}(h) = 441879.36\left[\frac{3}{2}\left(\frac{h}{18.68}\right) - \frac{1}{2}\left(\frac{h}{18.68}\right)^3\right] + 31525.04$$



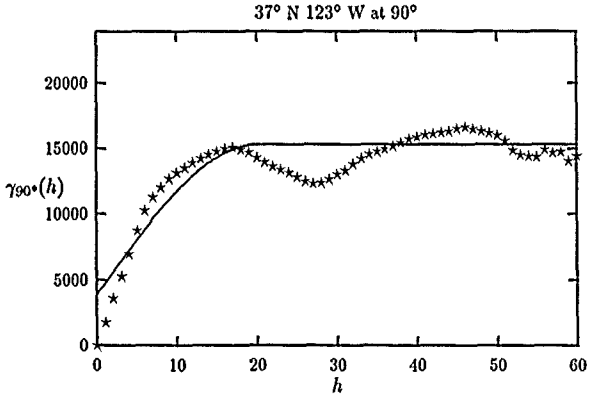
$$\gamma_{0°}(h) = 16764.17\left[\frac{3}{2}\left(\frac{h}{28.07}\right) - \frac{1}{2}\left(\frac{h}{28.07}\right)^3\right] + 21079.78$$



$$\gamma_{90°}(h) = 22489.87\left[\frac{3}{2}\left(\frac{h}{23.89}\right) - \frac{1}{2}\left(\frac{h}{23.89}\right)^3\right] + 14157.72$$



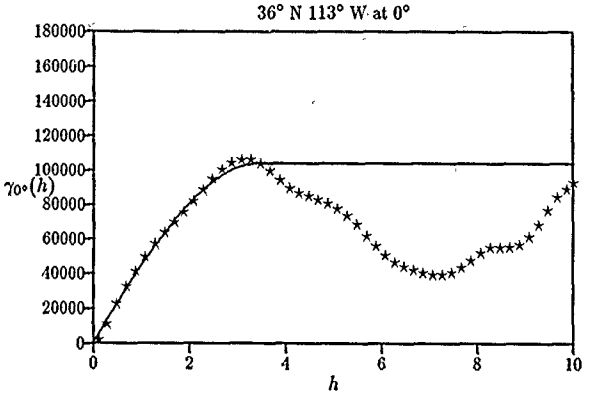
$$\gamma_{0°}(h) = 16406.05\left[\frac{3}{2}\left(\frac{h}{21.49}\right) - \frac{1}{2}\left(\frac{h}{21.49}\right)^3\right] + 2268.28$$



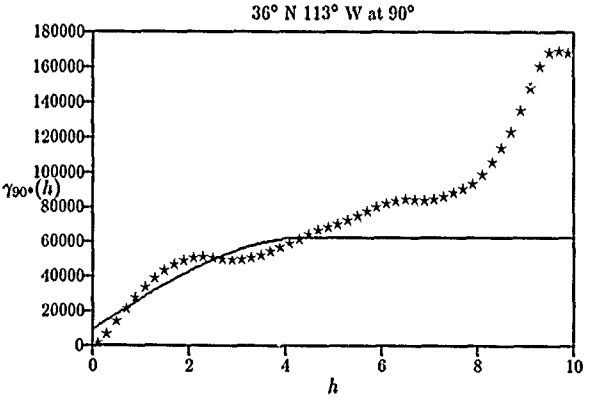
$$\gamma_{90°}(h) = 11340.39\left[\frac{3}{2}\left(\frac{h}{19.94}\right) - \frac{1}{2}\left(\frac{h}{19.94}\right)^3\right] + 3915.36$$

C.4 *Variograms for the Partial DTED Cells Used in Chapter V*

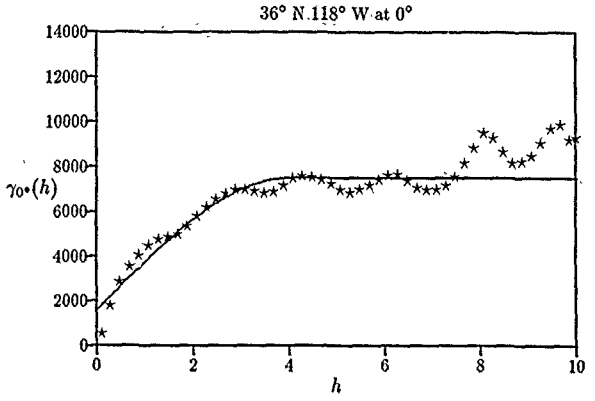
The following are directional experimental variograms and spherical variogram models as generated by `varfit` from the portions of the DTED Test Cells used in Chapter V. The first partial cell is from DTED Test Cell #1 (36°20′N ×112°30′W to 36°30′N ×112°40′W), while the second partial cell is from DTED Test Cell #2 (38°20′N ×120°30′W to 38°30′N ×120°40′W). These experimental variograms and variogram models were generated by `varfit` on *residual* terrain elevation data from the partial cells; the global trend was removed using `resid`. The latitude and longitude of the southwest corner of the DTED cell from which each partial cell was taken is provided above the variogram. Also indicated is the angle at which the experimental variogram was generated (in degrees clockwise from north) and the equation for the spherical model fitted to the experimental variogram by `varfit`. The experimental variograms are denoted by the stars (⋆), while the spherical variogram models are denoted by solid lines.



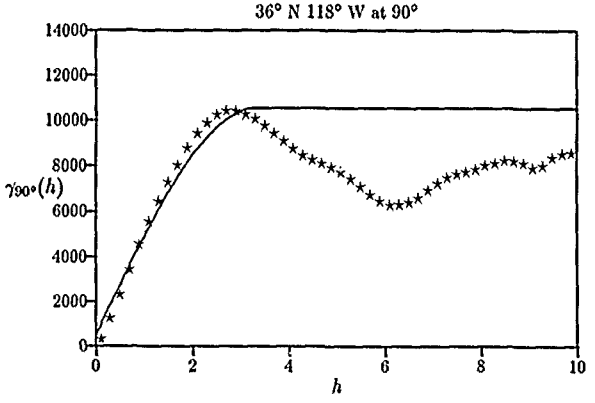
$$\gamma_{0°}(h) = 102516.70\left[\frac{3}{2}\left(\frac{h}{3.41}\right) - \frac{1}{2}\left(\frac{h}{3.41}\right)^3\right] + 1372.92$$



$$\gamma_{90°}(h) = 52757.31\left[\frac{3}{2}\left(\frac{h}{4.39}\right) - \frac{1}{2}\left(\frac{h}{4.39}\right)^3\right] + 9561.79$$



$$\gamma_{0°}(h) = 5899.53\left[\frac{3}{2}\left(\frac{h}{3.93}\right) - \frac{1}{2}\left(\frac{h}{3.99}\right)^3\right] + 1593.93$$



$$\gamma_{90°}(h) = 9977.56\left[\frac{3}{2}\left(\frac{h}{3.30}\right) - \frac{1}{2}\left(\frac{h}{3.30}\right)^3\right] + 558.95$$

Appendix D. .pts *File Format*

The .pts file format was created to facilitate the development, testing, and use of the terrain modeling pipelines described in this thesis. A .pts file is an ASCII file that contains xyz tuples, where z represents the terrain elevation at a specific x, y position. Each file also contains a header with control information for the various programs in the modeling pipelines. These files are usually created by `dted2pts` based on DMA DTED (see Chapter IV). The following sections describe the .pts file data and header formats in greater detail.

D.1 .pts *File Data*

All data in a .pts file should be in the format of xyz tuples where x, y, and z are floating point numbers. Each value in the tuple should be separated by white space (a space, tab, or carriage return); commas are not allowed. The tuples themselves should also be separated by white space. The tuples must immediately follow the `end_of_header` keyword described below; no additional control information must be embedded amongst these tuples.

Most programs in the modeling pipelines expect these tuples to be regularly spaced and to fall on the vertices of a regular grid as specified in the .pts file header. They also expect the gridded tuples to be arranged in column-major order (the x values should change more rapidly than the y values). The programs that require this gridded format will differentiate between .pts files with gridded data and .pts files with irregularly spaced data by the presence (or absence) of the keywords `xpts`, `ypts`, and `pts` in the header as described below. It is important to note that x and y should not specify the row and column of the corresponding gridded data point but should indicate the true x and y position of the data point. Files with gridded data should specify `xrange` and `yrange` in their header as described below so that the rows and columns corresponding to the xyz tuples can be computed.

D.2 .pts File Header

The .pts file format uses a header to carry general control information between the programs in the modeling pipelines. Each piece of information in the .pts file header consists of a keyword followed by one or more values. The characters in the keywords may be in mixed upper and lower case. The programs in the terrain modeling pipelines will ignore any unrecognized keyword and any values without preceding keywords that may be in the header. Each keyword and every value associated with that keyword must be followed by white space (a space, tab, or carriage return). This means that more than one keyword-value pair can be placed on the same line or a keyword can be separated from its associated value(s) by a carriage return. The keyword **end_of_header** indicates the end of the control information; the rest of the file is assumed to be data, whose format is described above. The ordering of the information in the header, with the exception of the end_of_header entry, is unimportant. No entry is required in the .pts file header with the exception of **end_of_header**; however, it is recommended that at least a **pts** entry or a **xpts** and **ypts** pair be used (see their descriptions below).

When using the modeling pipelines described in this thesis, most header entries are placed in the .pts file header by **dted2pts**; this is the first program in the modeling pipelines that creates a .pts file. Other header entries are added by the other programs as they generate the information needed to complete them. Once a header entry is added, all other programs in the modeling systems pass this entry on to their output .pts file unchanged unless the particular program was designed to modify that entry.

The following are the acceptable entries in a .pts file header. In the following descriptions, boldface words are keywords, *i* indicates an integer number, *f* indicates a floating point number, and a forward-slash (/) between two entries signifies that one or the other must be used.

pts *i*

This entry indicates the number of *xyz* tuples that follow the **end_of_header** keyword. Most programs in the modeling pipelines recognize this entry as meaning that the tuples are in an irregular order and are not gridded. The value associated with

this keyword is used to read the proper number of tuples from the .pts file and to allocate storage for these tuples. This entry should not be used with the xpts and ypts entries described below.

xpts i_x ypts i_y

These entries, which should always be used together, indicate that the xyz tuples that follow the end_of_header keyword are regularly spaced on a $i_x \times i_y$ grid and are ordered in column-major order. There should be a total of $i_x \times i_y$ tuples in the file. The values i_x and i_y are used to read the proper number of tuples from the .pts file and to allocate storage for these tuples. This entry should not be used with the pts entry described above.

xrange f_{xl} f_{xh} yrange f_{yl} f_{yh}

These entries indicate the lowest and highest x and y values among all the xyz tuples in the .pts file. These entries are required if the data is gridded to indicate the the actual size of the grid and to compute the grid row and column for each tuple.

xstep f_x ystep f_y

These entries indicate the distance between any two adjacent points on the grid that is defined by xpts, ypts, xrange, and yrange as described above. These entries would only be used if the data is regularly gridded, i.e., xpts and ypts is also specified. The step information provided by this entry is redundant since it can be calculated from the other entries, but most programs in the modeling pipelines expect it anyhow.

originLat f_y N/S originLong f_x E/W

These entries indicate the latitude and longitude corresponding to the origin from which all xyz tuples are based $(x, y = 0, 0)$. The longitude and latitude should be specified in degrees; the fractional part of the floating point number does *not* represent minutes or seconds. For gridded data generated by dted2pts, these latitude and longitude entries always corresponds to the southwest corner of the grid. All programs in the modeling pipelines assume that the x coordinate of any xyz tuple corresponds with longitude, while the y coordinate corresponds with latitude.

variogram_step *f*

This entry provides a stepsize for use by `varfit` for computing variograms (see Chapter IV). This stepsize is set by default to 1.0 in `.pts` files created by `dted2pts`.

variogram_regularization_angle *f*

This entry provides a variogram regularization angle for use by `varfit` for computing variograms (see Chapter IV). This entry should be in degrees. It is set by default to 1.0 in `.pts` files created by `dted2pts`.

variogram_maxlag *i*

This entry provides a variogram maximum lag for use by `varfit` for computing variograms (see Chapter IV). This maximum lag is set by default to 100 in `.pts` files created by `dted2pts`.

trend_polynomial $f_0\ f_1\ f_2\ f_3\ f_4\ f_5$

This entry is added by `resid` to provide the global trend polynomial coefficients to `rebuild` so that the global trend extracted by `resid` can be replaced (see Chapter IV). These values correspond to coefficients of a 2D quadratic polynomial of the following form:

$$M_{\mathtt{global}} = f_0 + f_1 x + f_2 y + f_3 xy + f_4 x^2 + f_5 y^2 \qquad \text{(D.1)}$$

spherical_coefficients *i* $f_A\ f_C\ f_{C0}\ f_q$

This entry is added by `varfit` to provide spherical variogram models to `krige` (see Chapter IV). The integer *i* corresponds to the angle that the specific variogram was generated. The next three values are coefficients of the spherical variogram model as fitted to the experimental variogram generated by `varfit`. This model is of the form:

$$\gamma(h) = \begin{cases} f_C\left(\frac{3}{2}\frac{h}{f_A} - \frac{1}{2}\frac{h^3}{f_A^3}\right) + f_{C0} & \text{if } h < f_A \\ f_C + f_{C0} & \text{if } h \geq f_A \\ 0 & \text{if } h = 0 \end{cases} \qquad \text{(D.2)}$$

The value f_q corresponds to a simple correlation of this model to the experimental variogram as determined by `varfit`.

linear_coefficients i f_{B0} f_{B1} f_q

This entry is added by `varfit` to provide linear variogram models to `krige` (see Chapter IV). The integer i corresponds to the angle that the specific variogram was generated. The next two values are coefficients of the linear variogram model as fitted to the experimental variogram generated by `varfit`. This model is of the form:

$$\gamma(h) = f_{B0}h + f_{B1} \tag{D.3}$$

The value f_q corresponds to a simple correlation of this model to the experimental variogram as determined by `varfit`.

dewijsian_coefficients i f_{B0} f_{B1} f_q

This entry is added by `varfit` to provide DeWijsian variogram models to `krige` (see Chapter IV). The integer i corresponds to the angle that the specific variogram was generated. The next two values are coefficients of the DeWijsian variogram model as fitted to the experimental variogram generated by `varfit`. This model is of the form:

$$\gamma(h) = f_{B0}\ln h + f_{B1} \tag{D.4}$$

The value f_q corresponds to a simple correlation of this model to the experimental variogram as determined by `varfit`.

iax i_x iay i_y

This entry provides IAX and IAY values to `krige` to define the kriging neighborhood as described in Chapter IV. `krige` will also accept these parameters from its command line. `krige` will write these values to its output `.pts` whether or not they were read from the input `.pts` file in order to indicate the values it used for IAX and IAY. No other program in the modeling pipeline places these entries in the `.pts` file.

minutes meters

These entries are included in `.pts` files created by `dted2pts` to document that the x and y units are arc minutes of latitude or longitude (approximately 1 nautical mile), while the z units are meters. Although all programs in the modeling pipelines pass these entries on from their input `.pts` files to their output `.pts` files, they are effectively ignored.

tolerance *f*

This entry is included in `.pts` files created by `krige` to document the tolerance used while kriging; points whose separation is within this tolerance were considered as being at the same location for the purpose of kriging (see Chapter IV). As this entry is placed in an output `.pts` file by `krige` solely for documentation purposes, *no* program in the modeling pipelines recognizes this entry in an input `.pts` file.

maximum_variance *f* **largest_difference** *f*

These entries are included in `.pts` files created by `krige` to document the maximum error variance or largest difference used by `krige` while performing minimization with respect to maximum error variance or largest difference (see Chapter IV). As these entries are placed in an output `.pts` file by `krige` solely for documentation purposes, *no* program in the modeling pipelines recognizes these entries in an input `.pts` file.

end_of_header

This entry simply indicates the end of the header and the start of the xyz tuples. It is the only mandatory entry in the header of the `.pts` file, and it must be the last entry in the header.

Appendix E. *User's Manuals*

The following pages are `Unix` man pages for the various programs written during the course of this thesis. More complete documentation can be found in Chapter IV. As `krige` was written by Brodkin (2), additional documentation for this program can be found there.

NAME

connect - creates database files for terrain model building.

SYNOPSIS

connect -o databasefile -g geomfile [-p ptsfile] [-d #divisions] [-l L/M/H] [-t t1 t2 t3 t4] [-s sX sY sZ] [-h]

DESCRIPTION

Connect creates the database and link files (collectively known as the database files) needed to create a terrain model to be used by systems using the Graphical Database Management System. These database files provide various connectivity and level of detail information for a polygonal terrain model. The primary purpose of these files is to allow multiple polygonal descriptions in different AFIT Geometry Files to be used together, either as different terrain grids in a large area or as different levels of resolution of the same grid.

Connect builds the database files in an incremental fashion. Provided with the name of a .pts file and the AFIT Geometry Files that were created from it by pts2geom , this program adds the necessary information from these files into the database files; the database files are created if they do not already exist. The input .pts file is read from stdin if not specified. The base name of the database files must always be specified; these filenames are differentiated only by their .lnk and .dbs extensions.

As the specified AFIT Geometry File names are simply added to the database files without checking for their presence, they do not need to exist until the database files are used to render them. Also, connect writes the input .pts file to stdout without modification if (and only if) it was read from stdin These facts allow connect to be run before pts2geon in the Unix terrain modeling pipeline; this is important since pts2geom outputs AFIT Geometry Files instead of .pts files.

The terrain model being built by progressively adding new grid squares is rearranged after each addition so that its origin (x,y = 0,0) is always at the center of the entire model.

OPTIONS

-o databasefile
The base name of the database files to be built or added to. This parameter is required.

-p ptsfile
The name of the .pts file containing the terrain

elevation data that is to be used to build the terrain models. If not specified, this file is read from stdin and written unmodified to stdout in support of the Unix terrain modeling pipeline.

-g geomfile
Specifies the name of the single AFIT Geometry File created by pts2geom from the specified .pts file, or if multiple AFIT Geometry Files were created from this specified .pts file, then it specifies the AFIT Geometry File name format in the same fashion as specified to pts2geom when created.

AFIT Geometry Files that are different sizes cannot be added to the same database files.

This option must be identical to the -o option specified to pts2geom when using the modeling pipelines.

-d #divisions
Specifies the number of divisions that pts2geom made to the gridded .pts file data when creating AFIT Geometry Files. If this option is omitted, it is assumed that no division of the terrain data was performed.

AFIT Geometry Files that were built using connect with a different number of divisions than files that are already present in the database files cannot be added.

This option must be identical to these same options specified to pts2geom when using the modeling pipelines.

-l L/M/H
Specifies the level of detail (low, medium, or high) that the terrain model being added to the database files corresponds to. This information is simply placed in the database files; it does not affect anything else.

-t t1 t2 t3 t4
Specifies the distances that the renderer should transition between models at different levels of detail. The first two values, t1 and t2, specify the distances from the viewer between which the low and medium level models are blended, while t3 and t4 specify the distances from the viewer between which the medium and high level models are blended. If omitted, no transition distances are added to the database files and the default transition distances of the rendering system will be used.

These values must match any transition distances already added to the database files.

-s sX sY sZ
Specifies a scaling value to pass on to the rendering system. This scaling should be applied to all models in the database files by the rendering system.

-h Prints help page and exits.

SEE ALSO
tape2dted(AFIT), dted2pts(AFIT), resid(AFIT), varfit(AFIT), krige(AFIT), rebuild(AFIT), partition(AFIT), pts2geom(AFIT), tb(AFIT), pts(AFIT)

Duckett, Donald P., The Application of Statistical Estimation Techniques to Terrain Modeling , MS Thesis, AFIT/GCE/ENG/91D-02, School of Engineering, Air Force Institute of Technology (AU), Wright-Paterson AFB OH, December 1991.

BUGS
No indication is given if connect is building a new set of database files or adding to existing ones. The user must remember to first delete existing database files before rebuilding them.

Connect creates temporary files names _temp.dbs and _temp.lnk. As unique temporary filenames are not used each time connect is run, it is not advisable to invoke more than one connect job at one time in the same directory.

NAME

dted2pts - creates a .pts file from DMA DTED.

SYNOPSIS

dted2pts [-i infile] [-o outfile] [-g gridsize] [-s N-limits S-limits E-limits W-limits] [-h]

DESCRIPTION

Dted2pts restructures the data in a DMA DTED file (.dtl file) into the .pts file format for further manipulation by other programs to build a polygonal terrain model. The data in the DTED file corresponding to the desired area will be filtered by sampling to produce this terrain description. The input file should be of the file format used on DTED CD-ROMs; this format is also created by tape2dted from DMA DTED 9 track tape. By default, input is read from stdin and output is written to stdout in support a Unix pipeline for terrain modeling. The data written to the .pts file is regularly gridded and is arranged in column-major form.

This program currently only supports Level 1 DTED.

OPTIONS

-i infile
Specifies the input DTED file. If omitted, input is read from stdin.

-o outfile
Specifies the output .pts file. If omitted, output is written to stdout.

-g gridsize
Specifies the spacing at which to sample the DTED to create the .pts file. This spacing should be specified in arc minutes of longitude or latitude (one arc minute of latitude = approximately one nautical mile). If data does not exist at the desired resolution in the DTED file, the desired spacing is reduced until it does correspond with the DTED spacing; if the desired spacing is below the finest resolution available in the DTED file, this minimum resolution will be used. For convenience, dted2pts considers all DTED elevation posts as separated by 0.05 arc minutes; DTED that does not maintain this post spacing (such as that above 40 degrees north latitude and below 40 degrees south latitude), would be interpreted by dted2pts as if it did.

-s N-limits S-limits E-limits W-limits
Specifies the area within the DTED file that will be filtered and converted into the .pts format. These limits must be specified in degrees of longitude or

latitude (without any hemisphere designator) and must be specified in the order shown (north, south, east, then west). The fractional part of the limiting latitudes and longitudes should be fractional degrees, not arc minutes of latitude or longitude. The elevation data within the desired limits must be entirely contained within the specified input DTED file.

-h Prints help page and exits.

SEE ALSO

tape2dted(AFIT), resid(AFIT), varfit(AFIT), krige(AFIT), rebuild(AFIT), partition(AFIT), connect(AFIT), pts2geom(AFIT), tb(AFIT), pts(AFIT).

Duckett, Donald P., The Application of Statistical Estimation Techniques to Terrain Modeling , MS Thesis, AFIT/GCE/ENG/91D-02, School of Engineering, Air Force Institute of Technology (AU), Wright-Paterson AFB OH, December 1991.

BUGS

As no hemisphere designator is allowed when the -s option is used to specify an area to be converted to the .pts format, the -s option will only work properly on DTED files from the Northern and Western Hemispheres.

NAME

krige - estimates elevations on a regular grid based on data from an input .pts file.

SYNOPSIS

krige [-i infile] [-o outfile] [-v variancefile] [-s #X #Y] [-m] [-a n] [-ld n] [-h]

DESCRIPTION

Krige accepts as input a standard .pts file that contains spherical variogram model parameters for both 0 and 90 degrees as generated by varfit. If this file is not specified, input data is read from stdin. This program can perform one of two basic functions: the kriging estimation of elevations at positions not provided in the input .pts file, or the selection of a minimized subset of elevation posts that represents the entire data set within a specified largest difference.

The estimation process is performed by estimating elevations based on the sampled data and spherical variogram model parameters provided through the input .pts file. This input data need not be gridded, but the estimates are always made at the vertices of a regular grid whose spacing is specified by the user. If vertices from this new grid coincide with points in the input .pts file, the elevation value at these points are used in place of the estimates; otherwise the kriging weighted sum estimator is used. The error variances of the estimates can also be written to a separate .pts file for additional analysis if desired.

The minimization process minimizes the data set by adding points from the original data set at positions where the difference between the actual elevation and the kriged estimate of the elevation is maximum; this is continued until this maximum difference is less than a given threshold called the largest difference.

This program uses universal kriging with linear drift in both of the above applications to estimate elevations at unknown positions and to compute an error variance associated with any estimation. The neighborhood used during estimation is defined by the range of the spherical variogram model; this range may be artificially constrained as described below to obtain quicker but less accurate estimates. The linear and De Wijsian variogram models, although provided in the .pts file by varfit, are not used by this program.

The neighborhood size used by krige to estimate an elevation can be artificially constrained to obtain a less then optimal estimate where an optimal estimate would be either

too time and CPU intensive. All elevation data in krige is stored in buckets corresponding to output grid vertices and these buckets are searched for points that may lie in a estimated point's neighborhood by starting with the bucket corresponding with that estimated point and working outward. Krige allows the user to specify how many steps outward this bucket search may progress, but as a safeguard it automatically limits the number of steps to include not more than 250 points.

Geometric anisotropy is handled by krige by using an anisotropy ratio. The x component of the distance between the position with known value and the position to be estimated is scaled by k = y-range/x-range before using the distance to compute a semi-variance or covariance. Krige simply averages the two sills to avoid the problem of zonal anisotropy.

OPTIONS

-i infile
Specifies the input .pts file; defaults to stdin if not specified.

-o outfile
Specifies the output .pts file; defaults to stdout if not specified.

-v variancefile
Specifies the .pts file to place error variance information in. If not specified, no such file is created.

-s #X #Y
Specifies the grid resolution over which kriging estimates are made. This is a required parameter. This parameter should not be used if minimizing.

-m If specified, minimization is performed. The -ld option must also be specified in this case. Do NOT specify the -s option when minimizing. -a n Reduce the number of buckets search to determine point in a specific neighborhood (IAX and IAY) to a radius of n buckets around the estimated point. If n = 0, then only the bucket associated with the grid location being estimated is searched.

-ld n
Specifies the largest difference to allow when minimizing with respect to largest difference. Must be specified if minimizing (if the -m option is used).

-h Prints help page and exits.

SEE ALSO

tape2dted(AFIT), dted2pts(AFIT), resid(AFIT), varfit(AFIT), partition(AFIT), rebuild(AFIT), connect(AFIT), pts2geom(AFIT), tb(AFIT), pts(AFIT).

Duckett, Donald P., The Application of Statistical Estimation Techniques to Terrain Modeling , MS Thesis, AFIT/GCE/ENG/91D-02, School of Engineering, Air Force Institute of Technology (AU), Wright-Paterson AFB OH, December 1991.

BUGS

Krige requires enough memory to store all the terrain elevation data in the .pts file at one time. Some machines may not be able to handle this without thrashing.

Krige also solves for the kriging weights by solving a system of equations by matrix inversion. This inversion is performed by LU decomposition and backsubstitution. However, this inversion is VERY slow on large matrices that result from large neighborhood or dense terrain data. Also, it can be unstable for matrices with over 250 rows and columns.

NAME

partition - partitions terrain data in a .pts file into homogeneous rectangular regions.

SYNOPSIS

partition [-i infile] -o outfile [-s N] [-g] [-h]

DESCRIPTION

To support the program krige in the Unix terrain modeling pipelines described, partition attempts to handle zonal anisotropy by partitioning the terrain elevation data from the input .pts file into multiple output .pts files containing elevation data homogeneous with respect to their average elevation. The input .pts file must contain regularly-spaced gridded terrain elevation data arranged in column-major order. Rectangular homogeneous regions are extracted from the input .pts file in the following fashion:

1. Sum the z values (elevations) over every row and every column of the gridded elevation data.

2. Reset the values of the rows/columns sums to the average of the rows/columns sums within a window of width N around that particular row/column.

3. Determine the median of the row sums and the column sums.

4. Label each row/column as to whether its sum is higher or lower than the median.

5. Partition the data set along horizontal boundaries where the row sums transition from above the median to below the median or from below the median to above the median. Create vertical boundaries in a similar fashion with column sums.

If not specified, the input is read from stdin. However, each of the partitions are written to a separate output .pts file and cannot be written to stdout. These files are named according to a user-specified filename format.

As output cannot be written to stdout, this program, if used, breaks the Unix terrain modeling pipelines; multiple smaller pipelines may be re-started using each of the resulting .pts files.

OPTIONS

-i infile
Specifies the input .pts file. If not specified, input is read from stdin.

-o outfile

Specifies the format for naming the multiple output .pts files. This outfile must be of the form basename.ext; the multiple output .pts files will be named basename0.ext through basenameN.ext, where N = one less than total number of blocks resulting from the partitioning.

-s N Specifies the window size N as used in step 2 of the algorithm above.

-g Specifies that a pattern of characters are to printed to stdout indicating the partitioning pattern resulting from this program.

-h Prints help page and exits.

SEE ALSO

tape2dted(AFIT), dted2pts(AFIT), resid(AFIT), varfit(AFIT), krige(AFIT), rebuild(AFIT), connect(AFIT), pts2geom(AFIT), tb(AFIT), pts(AFIT).

Duckett, Donald P., The Application of Statistical Estimation Techniques to Terrain Modeling , MS Thesis, AFIT/GCE/ENG/91D-02, School of Engineering, Air Force Institute of Technology (AU), Wright-Paterson AFB OH, December 1991.

BUGS

None known.

NAME

pts2geom - creates an AFIT Geometry File based on the terrain elevation data in a .pts file.

SYNOPSIS

pts2geom [-i infile] [-o outfile] [-d #divisions] [-s x-scale y-scale z-scale] [-v] [-h]

DESCRIPTION

pts2geom processes terrain elevation data contained in a .pts file and creates one or more AFIT Geometry Files containing a polygonal terrain model based on this data. Although the .pts file format allows for gridded or ungridded data, pts2geom requires that the data in the input .pts file be regularly gridded and arranged in column-major form. The gridded data is converted into a regular triangular polygon description of the terrain in the AFIT Geometry File format with color and normal information. pts2geom can build one AFIT Geometry File from a .pts file, but it can also sub-divide a terrain description from a single .pts file into identically-sized grid squares, placing each in a separate AFIT Geometry File. These grid squares can be linked together by the program connect to produce a large terrain model that can be used with systems using the Graphical Database Management System.

pts2geom is designed as part of a Unix pipeline for terrain modeling. Therefore, pts2geom reads input data from stdin if an input .pts file is not specified. However, output is written to stdout only if no grid square division is desired; otherwise multiple output AFIT Geometry Files are created, one for each grid square, named in accordance to a user specified format. Since pts2geom is designed to be run as the last stage in the terrain modeling pipeline, not writing to stdout does not break the pipeline.

By default, the color of any resulting polygon is a factor of the z-value of its highest vertex after scaling; alternate triangular polygons are colored slightly different to create an illusion of texture. Polygons are colored white (rgb=(0.8, 0.8, 0.8) or rgb=(0.7, 0.7, 0.7)) if their highest z-value is at or above 3000 units; polygons with z-values between 0 and 3000 units are colored dull green (rgb=(0.62, 0.55, 0.0) or rgb=(0.7, 0.65, 0.0)); polygons with a z-value at 0 units are colored blue (rgb=(0.2, 0.2, 0.8) or rgb=(0.2, 0.2, 0.7)); polygons with z-values below 0 units are colored brown (rgb=(0.7, 0.4, 0.2) or rgb=(0.7, 0,3, 0,2)). The -v option can be used to specify the use of vertex colors.

AFIT Geometry Files built with pts2geom are always transitioned so that the origin (x,y = 0,0) is in the center of

the grid square contained within that file.

OPTIONS

-i infile
Specifies the input .pts file. If omitted, input is read from stdin.

-o outfile
Specifies the single output AFIT Geometry File or the format for naming multiple AFIT Geometry Files. Multiple files are created when the -d option is used and the number of divisions is greater than one. In this case, outfile must be of the form basename.ext; the multiple output files will be named basename0.ext through basenameN.ext, where N = one less than total number of blocks resulting from the divisions.

If used in the modeling pipelines, this option must be identical to the -g option for the connect program.

-d #divisions
Specifies the number of divisions to make to the gridded .pts file data. If only one division is specified, pts2geom acts as if this option was omitted. However, for N divisions where N>1, the gridded data is divided into N*N equally sized grid squares (N divisions in the x direction and N divisions in the y direction) and each gridsquare is processed separately and placed into its own AFIT Geometry File as described above. If this option is omitted, no division of the terrain data is performed.

This option must be the same as the -d option specified for connect when using the terrain modeling pipelines.

-s x-scale y-scale z-scale
Specifies the scaling factor to apply to the x, y and z values in the .pts file before creating the AFIT Geometry File(s). If omitted, all scaling defaults to 1.

-v Specifies to use vertex colors instead of polygon colors. Default is to use polygon colors. If this option is specified, a color is applied to each polygon vertex with the expectation that the rendering system will blend the colors across the polygon surfaces. A vertex is colored white (rgb=(0.9,0.9,0.9)) if their z-value is above 3000 units, blue (rgb=(0.2,0.2,0.7)) if its z-value is zero units, and brown (rgb=(0.7,0.4,0.1)) if its z value is below zero units. If the vertex's elevation falls between zero and 3000 units, the color is determined by the following

formula:

red = 0.9*(z/3000)

green = 0.3+0.6*(z/3000)**2

blue = 0.9*(z/3000)**3

A random number between 0.0 and 0.1 is added to each color component to provide some texture.

-h Prints help page and exits.

SEE ALSO

tape2dted(AFIT), dted2pts(AFIT), resid(AFIT), varfit(AFIT), krige(AFIT), rebuild(AFIT), partition(AFIT), connect(AFIT), tb(AFIT), pts(AFIT).

Duckett, Donald P., The Application of Statistical Estimation Techniques to Terrain Modeling , MS Thesis, AFIT/GCE/ENG/91D-02, School of Engineering, Air Force Institute of Technology (AU), Wright-Paterson AFB OH, December 1991.

BUGS

None known.

NAME

rebuild - replaces global trend extracted by resid from terrain elevation data in a .pts file.

SYNOPSIS

rebuild [-i infile] [-o outfile] [-h]

DESCRIPTION

Rebuild is used in conjunction with krige and resid in the Unix terrain modeling pipelines to that the terrain elevation data used to produce kriged estimates are globally stationary. This program adds global trend that has been removed by resid back to residual terrain elevation data. The coefficients from a two-dimensional second-order polynomial is read from the input .pts file header and are added back to the elevation data in that .pts file. The second order polynomial used is of the form:

$$M = a_0 + a_1*x + a_2*y + a_3*x*y + a_4*x**2 + a_5*y**2$$

The trend polynomial coefficients are reset to zero before being written back to the header of the output .pts file.

If rebuild is run on a .pts file with no global trend information in its header, no changes are made to the data in the .pts file.

In support of the Unix pipeline, input is read from stdin if no input .pts file is specified, and output is written to stdout if no output .pts file is specified.

OPTIONS

-i infile
Specifies the input .pts file. If not specified, input is read from stdin.

-o outfile
Specifies the output .pts file. If not specified, output is written to stdout.

-h Prints help page and exits.

SEE ALSO

tape2dted(AFIT), dted2pts(AFIT), resid(AFIT), varfit(AFIT), krige(AFIT), partition(AFIT), connect(AFIT), pts2geom(AFIT), tb(AFIT), pts(AFIT).

Duckett, Donald P., The Application of Statistical Estimation Techniques to Terrain Modeling , MS Thesis, AFIT/GCE/ENG/91D-02, School of Engineering, Air Force Institute of Technology (AU), Wright-Paterson AFB OH, December 1991.

BUGS

None known.

NAME

resid - removes global trend from terrain elevation data in a .pts file.

SYNOPSIS

resid [-i infile] [-o outfile] [-p polynomialfile] [-h]

DESCRIPTION

Resid is used in conjunction with krige and rebuild to insure that the terrain elevation data used to produce kriged estimates is globally stationary. This program accepts terrain elevation data from a .pts file and extracts global trend by fitting a two-dimensional second-order polynomial to the data using least-squares regression. The second order polynomial used is of the form:

M = a_0 + a_1*x + a_2*y + a_3*x*y + a_4*x**2 + a_5*y**2

This polynomial is subtracted from the elevation data before this data is written back out to the output .pts file. The coefficients of the polynomial fitted to the data are written to the header of the output .pts file; rebuild later uses this information to add the polynomial back to the kriged elevation data.

In support of the Unix pipeline, input is read from stdin if no input .pts file is specified, and output is written to stdout if no output .pts file is specified.

Because of the computational cost of fitting a second-order polynomial to a large data set, a pre-determined polynomial may be extracted from the data instead. In this case a second .pts file should be specified with the -p option. This second file should already contain polynomial trend coefficients in its header information. It is this polynomial that is extracted from the data contained in the first .pts file; no polynomial fitting to the input data is actually performed. The coefficients in the header of the second file could be placed there by hand, but the expectation is that this second data set is a filtered version of the first (created by specifying a larger gridsize when running dted2pts) that has already been processed by resid.

OPTIONS

-i infile
Specifies the input .pts file. If not specified, input is read from stdin.

-o outfile
Specifies the output .pts file. If not specified, output is written to stdout.

-p polynomialfile
Specified a .pts file with an already-fitted trend polynomial to be extracted from the elevation data in the input .pts file. If omitted, a global trend polynomial is fitted to the data in the input .pts file.

-h Prints help page and exits.

SEE ALSO
tape2dted(AFIT), dted2pts(AFIT), varfit(AFIT), krige(AFIT), rebuild(AFIT), partition(AFIT), connect(AFIT), pts2geom(AFIT), tb(AFIT), pts(AFIT).

Duckett, Donald P., The Application of Statistical Estimation Techniques to Terrain Modeling , MS Thesis, AFIT/GCE/ENG/91D-02, School of Engineering, Air Force Institute of Technology (AU), Wright-Paterson AFB OH, December 1991.

BUGS
None known.

NAME

tape2dted - reads DMA DTED 9 track tape and produces DTED files in the DMA DTED CD-ROM format.

SYNOPSIS

tape2dted [-i tapename] [-o outfile] [-l Lat Lat-Hemisphere Long Long-Hemisphere] [-a] [-h]

DESCRIPTION

Tape2dted reads DTED from an Unix device such as a 9-track tape drive and produces a binary data file of the next DTED file read from that device. Tape2dted creates output files in the DTED CD-ROM file format (.dtl files), which are readable by the next program in the terrain modeling pipelines, dted2pts.

Tape2dted assumes that the tape contains binary DTED files formatted as specified by DMA; each file should contain gridded terrain elevation data for a 1 degree by 1 degree DTED cell. This program reads the next such file encountered from the current tape position and writes the elevation information to the specified output .dtl file (or to stdout in the .dtl file format if no output .dtl file is specified). Alternately, it can be instructed to read every file from the current tape position to the end of the tape; in this case the output is written to .dtl files with pre-determined names based on their latitude and longitude. As this later case creates multiple output files, it breaks the terrain modeling pipelines.

OPTIONS

-i tapename
Specifies the Unix device name to read the DTED files from. Defaults to /dev/nrmt1h.

-o outfile
Specifies the output .dtl file. Defaults to stdout. Ignored if the -a option is used.

-l Lat Lat-Hemisphere Long Long-Hemisphere
If specified, the label on the next DTED file on the tape device is compared to this latitude and longitude (DTED files are labeled by the latitude and longitude of their southwest corners). If the next file does not match, the program exits with an exit code of 2. This is the only case when an exit code of 2 is used in this program. This is useful in a shell script to loop until the proper file is found on the tape; the exit code is 0 when the file is found.

-a Specifies that every file on the DTED tape from the current should be read and written to a separate dtl

file. The files are named with the following format: aXXbYYY.dtl, where a is either n or s referring to the northern or southern hemisphere, XX is the latitude in whole degrees (always two digits), b is either e or w referring to the eastern or western hemisphere, and YYY is the longitude in whole degrees (always three digits). The -o option is ignored in this case, and nothing is written to stdout.

-h Prints help page and exits.

SEE ALSO

dted2pts(AFIT), resid(AFIT), varfit(AFIT), krige(AFIT), partition(AFIT), rebuild(AFIT), connect(AFIT), pts2geom(AFIT), tb(AFIT), pts(AFIT).

Duckett, Donald P., The Application of Statistical Estimation Techniques to Terrain Modeling , MS Thesis, AFIT/GCE/ENG/91D-02, School of Engineering, Air Force Institute of Technology (AU), Wright-Paterson AFB OH, December 1991.

BUGS

None known.

NAME

varfit - generates a semivariogram based on the terrain data in a .pts file.

SYNOPSIS

varfit [-i infile] [-o outfile] [-p plotfile] [-v variogramfile] [-a variogram angle] [-s variogram stepsize] [-h]

DESCRIPTION

Varfit generates directional experimental variograms and variogram model parameters based on data read from the specified input .pts file to support the program krige and the terrain modeling pipelines. Varfit can generate variograms for either regularly gridded and irregularly spaced data. Experimental variograms are generated for both the northerly (0 degrees) and easterly (90 degrees) directions unless otherwise specified. If a variogram at a different direction is desired, only the single specific directional variogram specified is generated.

Varfit fits linear, De Wijsian, and spherical variogram models to the resulting experimental variograms using a least-squares regression method. The coefficients of these models are written to the header of the output .pts file. The angle that each particular variogram model is generated from and a simple correlation signifying how well each particular model fitted the experimental variogram are also written to the .pts file header. The remaining elevation data from the input .pts file is simply copied to the output .pts file for later use by krige. If no input .pts file is specified, the input is read from stdin; if no output .pts file is specified, output is written to stdout.

It should be noted that, even though three different models are fitted to the data, krige only uses the spherical models generated at 0 and 90 degrees to perform the desired estimations.

In addition to the output .pts file, varfit optionally produces a descriptive variogram file containing a listing of the experimental variograms and the variogram models generated, along with other information from the variogram fitting process. A plot file is also produced if desired; this file simply contains the x, y pairs plotting the experimental variogram.

OPTIONS

-i infile

Specifies the input .pts file; defaults to stdin if not specified.

-o outfile
Specifies the output .pts file; defaults to stdout if not specified.

-p plotfile
Specifies the name of the file to place the variogram x, y data into for later plotting by another program. If not specified, no plot file is created.

-v variogramfile
Specifies the name of the variogram file where descriptive information concerning the variograms generated from the data is written. If not specified, no variogram file is created.

-a variogram angle
Specifies the angle at which to generate the experimental variogram. If specified, this is the only angle used. If not specified, variograms are generated for 0 and 90 degrees. This entry should be specified in degrees clockwise from north.

-s variogram stepsize
To quantify the data in the .pts file, varfit considers all distances within this stepsize to be the same. This effectively groups all pairs of points into discrete buckets. This stepsize defaults to 1 if not specified.

-h Prints help page and exits.

SEE ALSO
tape2dted(AFIT), dted2pts(AFIT), resid(AFIT), krige(AFIT), partition(AFIT), rebuild(AFIT), connect(AFIT), pts2geom(AFIT), tb(AFIT), pts(AFIT).

Duckett, Donald P., The Application of Statistical Estimation Techniques to Terrain Modeling , MS Thesis, AFIT/GCE/ENG/91D-02, School of Engineering, Air Force Institute of Technology (AU), Wright-Paterson AFB OH, December 1991.

BUGS
None known.

Bibliography

1. Babiak, Nicholas J. and Carol S. Lydic. "The Defense Mapping Agency and Tomorrow's Advanced Aerospace Warfare Systems." In *NAECON 1990*, pages 260–4, 1990.
2. Brodkin, Chris. *The Application of Kriging for Controlled Minimization of Large Data Sets*. MS thesis, AFIT/GA/ENY/91D-14, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1991.
3. Brunderman, John A. *Design and Application of an Object-Oriented Graphical Database Management System for Synthetic Environments*. MS thesis, AFIT/GA/ENG/91D-01, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1991.
4. Carrera, Jesus and Ferenc Szidarovszky. "Numerical Comparison of Network Design Algorithms for Regionalized Variables," *Applied Mathematics and Computation*, *16*:189–202 (1985).
5. Clark, Isobel. *Practical Geostatistics*. London: Applied Science Publishers Ltd, 1979.
6. Costenbader, J. L. "CIG (Computer Image Generation) Data Bases in an Instance: Bits and Pieces." In *The IMAGE III Conference Proceedings*, pages 151–163, September 1984.
7. Cressie, Noel. "Kriging Nonstationary Data," *Journal of the American Statistical Association*, *81*(395):625–634 (September 1986).
8. Cressie, Noel. "Spatial Prediction and Ordinary Kriging," *Mathematical Geology*, *20*(4):405–421 (1988).
9. Cressie, Noel. "Geostatistics," *The American Statistician*, *43*(4):197–202 (November 1989).
10. Cressie, Noel. "The Origins of Kriging," *Mathematical Geology*, *22*(3):239–52 (1990).
11. David, Michael. *Geostatistical Ore Reserve Estimation*. New York: Eisevier Scientific Publishing Company, 1977.
12. Davis, John C. *Statistics and Data Analysis in Geology* (Second Edition). New York: John Wesley & Sons, 1986.
13. Defense Mapping Agency. *Digitizing the Future* (Third Edition). Product Brochure. St Louis.
14. Defense Mapping Agency, St Louis. *Defense Mapping Agency Products Specification for Digital Terrain Elevation Data (DTED)* (Second Edition), April 1986.
15. Evans and Sutherland. *Technical Overview: ESIG4000*. Product Brochure. Salt Lake City, Utah.
16. Filer, Robert Edward. *A 3-D Virtual Environment Display System*. MS thesis, AFIT/GCS/ENG/89D-2, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1989 (AD-A216 279).

17. Frederiksen, Poul and others. "A Review of Current Trends in Terrain Modeling," *ITC Journal*, *2*:101–106 (1985).

18. Gerken, Mark J. *An Event Driven State Based Interface For Synthetic Environments*. MS thesis, AFIT/GCS/ENG/91D-07, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1991.

19. Grant, Michael. *The Application of Kriging in the Statistical Analysis of Anthropometric Data Volume I*. MS thesis, AFIT/GOR/ENY/ENS/90M-8, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, May 1990 (AD-A220 613).

20. Harris, Frank E. and others. *Preliminary Work on the Command and Control Workstation of the Future*. Progress Report NPS52-88-027, Naval Postgraduate School, August 1988.

21. Henley, Stephen. *Nonparametric Geostatistics*. New York: John Wiley & Sons, 1981.

22. Huang, Yih-Ping. "Triangular Irregular Network Generation and Topographical Modeling," *Computers in Industry* (Neitherlands), *12*(3):203–13 (July 1989).

23. IIT Research Institute, Rome, NY. *Scenario User's/Operator's Manual*, June 1989.

24. Journel, A. G. "Kriging in Terms of Projections," *Mathematical Geology*, *9*(6):563–86 (1977).

25. Journel, Andre G. *Fundamentals of Geostatistics in Five Lessons*. Washington, D. C.: American Geophysical Union, 1989.

26. Kelley, Alex D. and others. "Terrain Simulation Using a Model of Stream Erosion," *Computer Graphics*, *22*(4):263–268 (August 1988).

27. Kennie, T. J. M. and R. A. McLaren. "Modeling for Digital Terrain and Landscape Visualization," *Photogrammetic Record (UK)*, *12*(72):711–45 (October 1988).

28. Kerbs, Lynda. "GEO-Statistics: The Variogram," *COGS Computer Contributions*, *12*(2):54–59 (August 1986).

29. Matheron, G. "Principles of Geostatistics," *Economic Geology*, *58*:1246–66 (1963).

30. McGee, Donald W. *The Application of Statistical Kriging to Improve Satellite Imagery Resolution*. MS thesis, AFIT/GSO/ENS/ENY/91D-13, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1991.

31. McGhee, Robert B. and others. *An Inexpensive Real-Time Interactive Three-Dimensional Flight Simulation System*. Summary Report NPS52-87-034, Naval Postgraduate School, August 1987 (AD-A184 340).

32. Miller, Gavin S. P. "The Definition and Rendering of Terrain Maps," *Computer Graphics*, *20*(4):39–48 (1986).

33. Mirante, Anthony and Nicholas Weingarten. "The Radial Sweep Algorithm for Constructing Triangulated Irregular Networks," *IEEE Computer Graphics & Applications*, *2*(5):11–21 (May 1982).

34. Oliver, Michael R. and others, "Interactive, Networked, Moving Platform Simulators," February 1988.

35. Olson, Robert A. *Techniques To Enhance the Visual Realism of a Synthetic Environment Flight Simulator.* MS thesis, AFIT/GCS/ENG/91D-16, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1991.

36. Philip, G. M. and D. F. Watson, "Comment on "Comparing Splines and Kriging"." A Letter to the Editor in *Computers & Geosciences*, 12(2):243-5 (1986). Published as a reply to Dubrule, O., "Comparing Splines and Kriging", *Computers & Geosciences*, 10(2-3):327-38 (1984).

37. Poplar, Jacquelyn and Reed Whittington. "Using Fractals to Create Realistic Three-Dimensional Terrain for the Training Environment." In *Proceedings of the 10th I/ITSC*, pages 203–208, 1988.

38. Press, William H. and others. *Numerical Recipes in C.* Cambridge: Cambridge University Press, 1988.

39. Rebo, Robert Keith. *A Helmet-Mounted Virtual Environment Display System.* MS thesis, AFIT/GCS/ENG/88D-17, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1988 (AD-A203 055).

40. Roberts, LeeAnne. *A Software System to Create a Hierarchical Multiple Level of Detail Terrain Model.* MS thesis, AFIT/GCS/ENG/88D-18, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1988 (AD-A203 047).

41. Robinson, D. Class notes from kriging seminar. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, 1991.

42. Ross, Harry D. *Analysis and Preparation of a Digital Terrain Data Base for Flight Simulator Use.* MS thesis, AFIT/GEO/MA/81D-1, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, November 1981 (AD-A115 547).

43. Simpson, Dennis J. *An Application of the Object-Oriented Paradigm to a Flight Simulator.* MS thesis, AFIT/GCS/ENG/91D-22, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1991.

44. Smith, Alvy Ray. "Plants, Fractals, and Formal Languages," *Computer Graphics*, *18*(3):1–10 (July 1984).

45 Southard, David A. *Superworkstations for Terrain Visualization: A Prospectus.* Technical Report MTR11080, Bedford, MA: The MITRE Corporation, January 1991.

46. Szeliski, R. and D. Terzopoulos. "From Splines to Fractals," *Computer Graphics*, *23*(3):51–60 (July 1989).

47. Szidarovszky, Ferenc. "Multiobjective Observation Network Design for Regionalized Variables," *International Journal of Mining Engineering*, *1*:331–42 (1983).

48. Szidarovszky, Ferenc and Sidney Yakowitz. *Principles and Procedures of Numerical Analysis.* New York: Plenum Press, 1978.

49. Van, J. K. "Advances in Computer Generated Imagery for Flight Simulation," *IEEE Computer Graphics & Applications*, *5*(8):37–51 (August 1985).

50. Veron, H. and others. *3D Displays for Battle Management*. Final Technical Report RADC-TR-90-46, The Mitre Corporation, April 1990.

51. Yakowitz, S. J. and F. Szidarovszky. "A Comparison of Kriging with Nonparametric Regression Methods," *Journal of Multivariate Analysis*, *16*:21–53 (1985).

52. Zyda, Michael J. and others. "Flight Simulators for Under $100,000," *IEEE Computer Graphics & Applications*, *8*(1):19–27 (January 1988).

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	December 1991	Master's Thesis

4. TITLE AND SUBTITLE
THE APPLICATION OF STATISTICAL ESTIMATION TECHNIQUES TO TERRAIN MODELING

5. FUNDING NUMBERS

6. AUTHOR(S)
Donald P. Duckett, Jr., Captain, USAF

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Air Force Institute of Technology, WPAFB OH 45433-6583

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT/GCE/ENG/91D-02

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Richard Slavinski
RL/COAA
Griffis AFB, NY 13441-5700

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

This thesis researches methods of generating accurate and realistic polygonal terrain models by reducing gridded sampled terrain elevation data such as DMA DTED. The terrain models generated should be applicable for use in flight simulators and other systems. Existing methods of terrain modeling are discussed, and limitations of these systems are presented.

The geostatistical estimation technique known as kriging is presented as a method of estimating terrain elevations at locations not provided in DMA DTED or other gridded terrain elevation data. Kriging is an optimal interpolation method based on statistical analysis of the elevation data. It also provides a measure of accuracy, the error variance, that allows some control over the accuracy of the resulting terrain model.

This estimation method is employed in a terrain modeling system that builds polygonal terrain models at any resolution. This system is described in this thesis, and the results of terrain modeled both by filtering the DMA DTED and by estimating elevations with kriging are presented. The technique as implemented is computationally very expensive and has therefore limited the results of this thesis effort. Also, terrain models that could be produced appeared smoother than their filtered counterparts. However, kriging still shows promise as a method of estimating terrain elevations for terrain models.

14. SUBJECT TERMS	15. NUMBER OF PAGES
Terrain Models, Computer Graphics; Flight Simulation; Statistical Analysis, Kriging	240
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

GENERAL INSTRUCTIONS FOR COMPLETING SF 298

The Report Documentation Page (RDP) is used in announcing and cataloging reports. It is important that this information be consistent with the rest of the report, particularly the cover and title page. Instructions for filling in each block of the form follow. It is important to ***stay within the lines*** to meet ***optical scanning requirements.***

Block 1. Agency Use Only *(Leave blank)*.

Block 2. Report Date. Full publication date including day, month, and year, if available (e.g. 1 Jan 88). Must cite at least the year.

Block 3. Type of Report and Dates Covered. State whether report is interim, final, etc. If applicable, enter inclusive report dates (e.g. 10 Jun 87 - 30 Jun 88).

Block 4. Title and Subtitle. A title is taken from the part of the report that provides the most meaningful and complete information. When a report is prepared in more than one volume, repeat the primary title, add volume number, and include subtitle for the specific volume. On classified documents enter the title classification in parentheses.

Block 5. Funding Numbers. To include contract and grant numbers; may include program element number(s), project number(s), task number(s), and work unit number(s). Use the following labels:

C	- Contract	PR	- Project
G	- Grant	TA	- Task
PE	- Program Element	WU	- Work Unit Accession No

Block 6. Author(s) Name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. If editor or compiler, this should follow the name(s).

Block 7. Performing Organization Name(s) and Address(es). Self-explanatory.

Block 8. Performing Organization Report Number Enter the unique alphanumeric report number(s) assigned by the organization performing the report

Block 9. Sponsoring/Monitoring Agency Name(s) and Address(es) Self-explanatory.

Block 10. Sponsoring/Monitoring Agency Report Number *(If known)*

Block 11. Supplementary Notes Enter information not included elsewhere such as: Prepared in cooperation with ; Trans of ; To be published in... When a report is revised, include a statement whether the new report supersedes or supplements the older report.

Block 12a. Distribution/Availability Statement. Denotes public availability or limitations. Cite any availability to the public. Enter additional limitations or special markings in all capitals (e.g. NOFORN, REL, ITAR).

- DOD - See DoDD 5230.24, "Distribution Statements on Technical Documents."
- DOE - See authorities.
- NASA - See Handbook NHB 2200.2.
- NTIS - Leave blank.

Block 12b. Distribution Code.

- DOD - Leave blank.
- DOE - Enter DOE distribution categories from the Standard Distribution for Unclassified Scientific and Technical Reports.
- NASA - Leave blank.
- NTIS - Leave blank.

Block 13. Abstract. Include a brief *(Maximum 200 words)* factual summary of the most significant information contained in the report

Block 14. Subject Terms. Keywords or phrases identifying major subjects in the report.

Block 15. Number of Pages. Enter the total number of pages.

Block 16. Price Code. Enter appropriate price code *(NTIS only)*.

Blocks 17. - 19. Security Classifications Self-explanatory. Enter U.S. Security Classification in accordance with U S Security Regulations (i e , UNCLASSIFIED). If form contains classified information, stamp classification on the top and bottom of the page.

Block 20. Limitation of Abstract This block must be completed to assign a limitation to the abstract. Enter either UL (unlimited) or SAR (same as report). An entry in this block is necessary if the abstract is to be limited. If blank, the abstract is assumed to be unlimited

*U S GPO 1990-0-273-271